HDL-TR-1715

A020493

Signal Analysis and Parametrization for EMP Data

September 1975

RESEARCH
DEVELOPMENT
ENGINEERING

U.S. Army Materiel Command
HARRY DIAMOND LABORATORIES
Adelphi, Maryland 20783

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER HDL-TR-1715	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Signal Analysis and Parametrization for EMP Data		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Thomas A. Tumolillo		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Harry Diamond Laboratories 2800 Powder Mill Road Adelphi, MD 20783		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Prog Ele 3.31.43K AMCMS Code 691000.22.10775 HDL Proj 220400
11. CONTROLLING OFFICE NAME AND ADDRESS Director Defense Nuclear Agency Washington, DC 20305		12. REPORT DATE September 1975
		13. NUMBER OF PAGES 163
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Commander National Military Command System Support Center Washington, DC 20305		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES
This work was sponsored by the Defense Nuclear Agency, MIPR #72-642, MIPR #74-597 and the National Military Command System Support Center MIPR #HC1001-4-00014; work unit title: Evaluation of EMP Effects of Mobile/Transporatable Communications Equipment.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Signal analysis
Digitized time series
Interpolation schemes
Digital filtering
Transform techniques
Nonlinear least squares
Computer program listings

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

    Presented are the theoretical background, program description, and computer listings that describe the techniques to reduce and analyze digital time-series data obtained in electromagnetic-pulse testing of military communications systems. The topics treated include data preprocessing, digital filtering, fast Fourier transform, fast Walsh transform, refined spectral densities, autocor-

DD $_{1 JAN 73}^{FORM}$ 1473   EDITION OF 1 NOV 65 IS OBSOLETE

relation functions, waveform representation by parametrized functions, and nonlinear, least squares techniques. Several examples of the data are analyzed and discussed.

CONTENTS

ILLUSTRATIONS

3

TABLES

1. INTRODUCTION

This report presents the theoretical background and computer programs for the analysis techniques that were used in evaluating data obtained in the electromagnetic pulse (EMP) testing of military communication and weapons systems under the PREMPT program.[1]

The data presented are initially obtained as a voltage-versus-time trace photographed on Polaroid film. This trace is then digitized and a time series of digital values is produced. The data are then processed in a digital computer. The various techniques employed in reducing and transforming the data are grouped under the generic title "signal analysis." Section 2 of this report gives a detailed description of all the algorithms; also, it contains complete instructions on how to use the signal-analysis program.

Another technique used in the data-reduction process is to represent the EMP waveform by a set of parametrized functions. This technique involves a least-squares fitting procedure which is discussed in section 3. Also, section 3 contains complete instructions on the use of a least-squares fitting program.

This report is not intended to be exhaustive on the subject of signal analysis but rather to present to the EMP community a basic software package that will: (1) accomplish most of the data reduction for EMP work and (2) be easily modified to include any additional techniques.

2. SIGNAL ANALYSIS

This section presents the theoretical background and computer implementation of a number of techniques for reducing and transforming digital time series produced in EMP tests under the PREMPT program. A complete program listing annotated with comments is given in this section. Several versions of this program have been implemented on both the IBM 370-195 and CDC 6500 computers. All of the programming was done in FORTRAN. Some of the subroutines have been coded in assembly language for the NOVA minicomputer but are not reported here.[2]

---

[1]The PREMPT program is a joint NMCSSC/DNA effort to determine the response of DCS to electromagnetic pulses generated by a high-altitude nuclear burst.

[2]Further details are presented in "The Interactive Digitization and Editing System (IDES)" by Dr. Thomas A. Tumolillo, USA Harry Diamond Laboratories, Adelphi, MD 20783, (Aug 1973).

2.1   Data Preprocessing

Under the generic title "Data Preprocessing" is included the many minute details that are necessary to prepare the raw input data as obtained from a digitization of the waveform, so that it is suitable for transformation to the frequency domain.

The following sections discuss the method of referencing the trace to the scope graticule, scaling the data, time ordering, bit reversal, and a few of the simpler interpolation schemes. The software for two of the simpler interpolators, the linear Lagrange, and the linear least squares are presented.   Higher order interpolators have been used occasionally in the PREMPT program, but are not included here because the simpler methods usually work. Simililarly, no techniques for time tying of digital records are given. Only the software are presented for the most commonly used grid and tablet referencing schemes, even though some of the more complicated procedures are discussed.

2.1.1   Referencing the Trace to Graticule and Digitization Tablet

One important function of the data reduction is the determination of the rotation angle of the scope graticule with respect to the digitization tablet, the zero point (origin) of the graticule, and the scale factors for the X and Y axes in the graticule (or grid) coordinate system. There are many possible schemes that can be used to determine these factors; the most general method will be discussed in this section.   The   software exists for the general procedure as well as for the simpler specific case implemented in this signal analysis package.

For simplicity, but with no loss of generality, let the grid points be symmetrical about the origin of the grid coordinate system (fig. 1); call this origin (X_o, Y_o). In the grid coordinate system the measure grid points are designated by the arrays XG(I) and YG(I).   In the tablet coordinate system the measured grid points are designated by the arrays XT(I) and YT(I).   The coordinate systems are shown in figure 1.

The transformation equation between the two systems is

$$
\begin{bmatrix} X_T \\ Y_T \end{bmatrix} = \begin{Bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{Bmatrix} \begin{bmatrix} X_G \\ Y_G \end{bmatrix} + \begin{bmatrix} X_o \\ Y_o \end{bmatrix} \tag{1}
$$
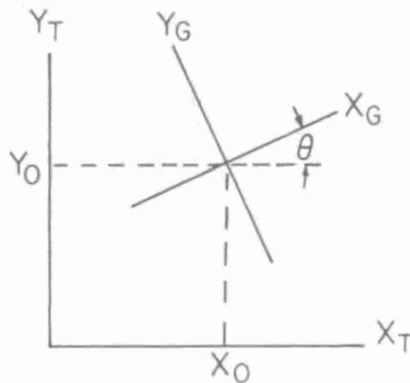
Figure 1. Diagram of coordinate systems.

Applying this transformation to the measured grid points, we have

$$
\begin{bmatrix} \sum\limits_{I=1}^{N} XT(I) \\[2em] \sum\limits_{I=1}^{N} YT(I) \end{bmatrix} = \left\{ \begin{matrix} \cos\theta & -\sin\theta \\[2em] \sin\theta & \cos\theta \end{matrix} \right\} \begin{bmatrix} \sum\limits_{I=1}^{N} XG(I) \\[2em] \sum\limits_{I=1}^{N} YG(I) \end{bmatrix} + \begin{bmatrix} \sum\limits_{I=1}^{N} X_o \\[2em] \sum\limits_{I=1}^{N} Y_o \end{bmatrix} \quad .
\tag{2}
$$

The points are symmetrical in the grid system; thus, we have

$$
\sum_{I=1}^{N} XG(I) = \sum_{I=1}^{N} YG(I) = 0 \quad .
$$

Therefore, $\left(X_o, Y_o \right)$ is given by

$$
\begin{bmatrix} X_o \\[2em] Y_o \end{bmatrix} = \frac{1}{N} \begin{bmatrix} \sum\limits_{I=1}^{N} XT(I) \\[2em] \sum\limits_{I=1}^{N} YT(I) \end{bmatrix}
\tag{3}
$$

9

It is convenient to shift the origin of the tablet coordinate system to the point $\left(X_O, Y_C\right)$; thus,

$$\begin{bmatrix} X'_T \\ Y'_T \end{bmatrix} = \begin{bmatrix} X_T - Y_O \\ Y_T - Y_O \end{bmatrix} = \begin{Bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{Bmatrix} \begin{bmatrix} X_G \\ Y_G \end{bmatrix} \tag{4}$$

For a specific set of points in the grid system, the above transformation can be linearized. Take the line defined by $Y_G = 0$; then we have

$$X'_T = \cos\theta \ X_G, \ Y_G = 0 \ .. \tag{5}$$

Let $\alpha = \cos\theta$, then $X'_T = \alpha X_G$. The best value of θ can be estimated from the measured points along the grid X axis by using a least-squares technique. Let

$$\chi^2 = \sum_I \ [XT(I) - \alpha \ XG(I)]^2 \ ; \tag{6}$$

minimizing χ^2 with respect to α, we have

$$\frac{\partial(\chi^2)}{\partial \ \alpha} = 0 \Rightarrow \alpha = \cos\theta = \frac{\displaystyle\sum_I XT(I)*XG(I)}{\displaystyle\sum_I XG(I)*XG(I)} \tag{7}$$

The $\displaystyle\sum_I$ means that we only sum over those points on the grid X axis.

We could now suitably define other straight lines on subsets of the measured grid points and get further estimates of θ. It is more convenient to use a nonlinear least-squares technique, and extract the best value of θ, by using all the points at once.

10

Let $\vec{Z}(I) = \begin{bmatrix} XT(I) \\ YT(I) \end{bmatrix}$, $\qquad\qquad$ (8)

$$\vec{\phi}(\theta,I) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{bmatrix} XG(I) \\ YG(I) \end{bmatrix} , \qquad (9)$$

and

$$\chi^2 = \sum_{I=1}^{N} |\vec{Z}(I) - \vec{\phi}(\theta,I)|^2 . \qquad (10)$$

We minimize χ^2 with respect to θ by requiring $\partial(\chi^2)/\partial\theta = 0$; thus, we have

$$\frac{\partial(\chi^2)}{\partial\theta} = \sum_{I=1}^{N} \left[\frac{\partial\vec{\phi}^{\dagger}(\theta,I)}{\partial\theta} \cdot (\vec{Z}(I) - \vec{\phi}(\theta,I)) + (\vec{Z}(I) \right.$$

$$\left. - \vec{\phi}^{\dagger}(\theta,I)) \frac{\partial\vec{\phi}}{\partial\theta}(\theta,I) \right] = 0. \qquad (11)$$

Here, $\phi^{\dagger}$ is the adjoint of ϕ.

Assume now that we have defined an iterative process for evaluating θ and $\vec{\phi}$, at the kth iteration, we assume that $\vec{\phi}$ is given by

$$\vec{\phi}(\theta,I) \sim = \vec{\phi}\left(\theta^k,I\right) + \frac{\partial\vec{\phi}\left(\theta^k,I\right)}{\partial\theta} \cdot \Delta\theta^k$$

$$\Delta\theta^k = \theta^{k+1} - \theta^k \qquad (12)$$

By substituting equation (12) into equation (11), a recursion relation is obtained for θ. It can be shown after a slight algebraic manipulation that

$$\theta^{k+1} = \theta^k - C_1 \sin\theta^k + C_2 \cos\theta^k , \qquad (13)$$

11

where

$$C_1 = \frac{\sum\limits_{I=1}^{N} (XG(I)*XT(I) + YG(I)*YT(I))}{\sum\limits_{I=1}^{N} (XG(I)*XG(I) + YG(I)*YG(I))} \quad , \quad (14)$$

$$C_2 = \frac{\sum\limits_{I=1}^{N} (XG(I)*YT(I) - YG(I)*XT(I))}{\sum\limits_{I=1}^{N} (XG(I)*XG(I) + YG(I)*YG(I))} \quad , \quad (15)$$

The recursion relation generally converges quickly, as long as the initial estimate for θ is close to the true value. We used for θ^1 the value determined in equation (7). The program considers that the iterative scheme has converged as long as $\Delta\theta^k < 10^{-3}$ and $\chi^{2(k+1)} < \chi^{2(k)}$.

The scale factors for the X and Y axes can be easily calculated. For illustrative purposes, suppose we measured 28 grid points as shown in figure 2.

Initially, the arrays XG and YG are defined in a DATA statement so that the points [XG(I), YG(I)] correspond to a grid so that

$$XG(I) \; \varepsilon \; \{-1.,-.8,-.6,\ldots,+.8,+1.\}$$

$$YG(I) \; \varepsilon \; \{-1.,-.75,\ldots,+.75,+1.\} \quad ,$$

so that for example [XG(27), YG(27)] = (+.4,-.5). In the program, one of the initial redefinitions of the arrays we make is

$$XG(I) \rightarrow XS*XG(I)$$

$$YG(I) \rightarrow YS*YG(I)$$

12
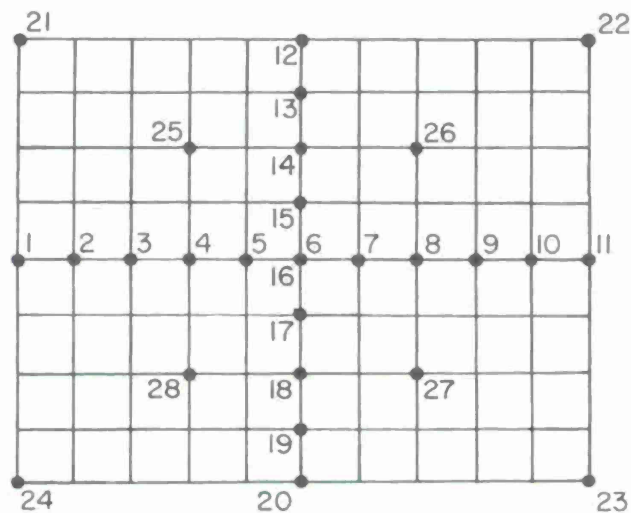
Figure 2. Measured grid points.

Scale factors XS and YS must be determined (fig. 3).
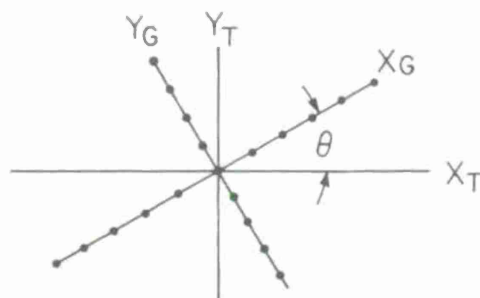
Figure 3. Rotation between G and T systems.

For the points on the X_G axis (I = 1,...,11), we have

$$[XS*XG(I)]^2 = XT(I)*XT(I) + YT(I)*YT(I)$$

The average value for XS is

13

$$XS = \sqrt{\dfrac{\displaystyle\sum_{I=1}^{11} (XT(I)*XT(I) + YT(I)*YT(I))}{\displaystyle\sum_{I=1}^{11} XG(I)*XG(I)}} \quad .\tag{16}$$

Similarly, from the points on the Y_G axis $(I = 12, 20)$, we have

$$YS = \sqrt{\dfrac{\displaystyle\sum_{I=12}^{20} (XT(I)*XT(I) + YT(I)*YT(I))}{\displaystyle\sum_{I=12}^{20} YG(I)*YG(I)}}\tag{17}$$

A complete grid measurement is rarely carried out as a routine operation in the reduction of EMP data. Occasionally, it will be done to test the linearity of the system. The most common method is to measure two points on each axis and calculate all quantities from these numbers. The following equations were implemented in the signal analysis package. The routine calculates the grid rotation angle, the scale factor, and the origin point from two measured points on both the X and Y grid axes. Initially, the program reads in the coordinates of the two points measured on the X axis—X1 and X2—and on the Y axis—Y1 and Y2. Then the four grid points are read in the same order that the coordinate locations were read in. The four grid points are stored in the arrays XT(I), YT(I), I = 1,4. The origin of the grid (X0, Y0) is given by

$$X0 = |X2*XT(1)-X1*XT(2)| \ / |X1-X2| \ ,$$

$$Y0 = |Y2*YT(3)-Y1*YT(4)| \ / |Y1-Y2| \ .\tag{18}$$

The scale factors XS and YS are given by

$$XS = \sqrt{(XT(2)-XT(1))**2 + (YT(2)-YT(1))**2} \ /(X2-X1) \ ,$$

$$YS = \sqrt{(XT(4)-XT(3))**2 + (YT(4)-YT(3))**2} \ / \ (Y2-Y1).\tag{19}$$

14

The rotation angle is determined by

$$\tan(\theta) = 0.5 * \left(\frac{YT(2)-YT(1)}{XT(2)-XT(1)} - \frac{XT(4)-XT(3)}{YT(4)-YT(3)} \right) \quad ,$$

$$CT = \cos(\theta) = 1./\sqrt{1+\tan(\theta)**2} \quad ,$$

$$ST = \sin(\theta) = \cos(\theta)*\tan(\theta) \quad . \tag{20}$$

Then the subroutine reads scale factors T (nanosecond/grid division) and V (volts/grid division), then recomputes the scale factors as

$$T \rightarrow T/XS, \quad V \rightarrow V/XS , \tag{21}$$

which have units nanoseconds/tablet counts and volts/tablet counts. Then the (x,y) coordinates of the trace are transformed as,

$$X(I) \leftarrow (CT*(X(I)-XO) + ST*(Y(I)-YO)-XZ)*T ,$$

$$Y(I) \leftarrow (-ST*(X(I)-XO) + CT*(Y(I)-YO)-YZ)*V , \tag{22}$$

where (XZ, YZ) are the rotated coordinates of the "zero point of the trace," that is, the point on the trace at which the signal starts,

$$XZ \leftarrow CT*(XZ-XO) + ST*(YZ-YO)$$

$$YZ \leftarrow -ST*(XZ-XO) + CT*(YZ-YO) \tag{23}$$

2.1.2 Time Ordering, Bit Reversal, and Interpolation Schemes

Time ordering of the array is a necessary procedure in order to remove errors introduced in the digitization process. Occasionally, there will be errors in the grid measurements that cause portions of the trace to fold back in the time sense after it is rotated. Similarly, inaccurate movement of the digitization operator's hand may also cause a few points to be folded back in the time sense. In some cases the digitization hardware will allow consecutive digital points to have the same time value. If these measurement ambiguities are not removed, they will cause considerable error in the high-frequency part of the transforms. This correction of the data is handled in subroutine CST OUT. This routine casts out those points in the array that are folded

15

back--that is, if XF is the input time array and XF(K)<XF(I) for K = I + 1, I + 2,... then SF(K) is deleted from the array. Similarly, if XF(K) = XF(I) for some set of K then the routine averages the amplitude YF(K) to create a single value at that value XF(I).

Bit reversal of the array refers to a specific reordering of the elements of a digital time series prior to its entering the FFT subroutine. It is done so that after transformation the frequency domain arrays are in ascending order of the frequency value. The term *bit reversal* arises from representing the index of an array I in base-two notation. For example, suppose we have the 65th element of a 1024 element array, then $65_{10} = 0001000001_2$. The reverse of the number is $1000001000_2 = 520_{10}$. To bit reverse, we swap the elements 65 and 520 of the original array.

In subroutine LNYQ, the Lagrangian methods for interpolation are used. A brief description of Lagrange interpolation is given here.

It is generally assumed that the function, f, interpolated here behaves like a polynomial; thus, in order to calculate f approximately at a point x, we find a polynomial approximation g for f good in the neighborhood of x. Lagrange showed that there is a unique polynomial of degree n having n + 1 values f_i at n + 1 distinct points x_i, i = 0,...n. That polynomial is g_n,

$$g_n(x) = \sum_{i=o}^{n} f(x_i) \prod_{\substack{j=o \\ j \neq i}} (x - x_j)/(x_i - x_j) \ . \tag{24}$$

For the software presented in this package, n is restricted to the value 1. Thus,

$$g_1(x) = \frac{f(x_1) - f(fx_0)}{x_1 - x_0} x + \frac{x_1 f(x_0) - x_0 f(x_1)}{x_1 - x_0}$$

$$\equiv C1 \cdot X + C2 \ . \tag{25}$$

In the program the function f is called YF(I), x_i is replaced by the time array XF(I), and the interpolated values are put into the real part of the complex array YNYQA(K)

```
REAL [YNYQA(K)] = Cl·X + C2,
```

$$Cl = \frac{YF(I)-YF(I-1)}{XF(I)-XF(I-1)} \quad , \quad C2 = \frac{XF(I)*YF(I-1)-XF(I-1)*YF(1)}{XF(I)-XF(I-1)} \quad . \quad (26)$$

In subroutine NYQST, a linear polynomial is fitted to a set of points in the arrays YF(I), XF(I), I = LB,...,LT. After determining the polynomial, the program evaluates it at the predetermined interpolation point.

The theory behind the least squares, polynomial fitting programs is straightforward. We need to minimize χ^2 with respect to the C(J) where

$$\chi^2 = \sum_{I=LB}^{LT} [YF(I) - \sum_{J=0}^{M} C(J)*XF(I)**J]^2 \quad . \quad (27)$$

Setting $\frac{\partial(\chi^2)}{\partial C(J)} = 0 \quad J = 0,...,M$ yields

$$\vec{B} = A \cdot \vec{C} \quad , \quad (28)$$

where the $(K,L)^{th}$ element of the matrix A is

$$A(K,L) = \sum_{I=LB}^{LT} XF(I)**(K + L - 2), \quad K, L = 1,...,M, \quad (29)$$

the Kth element of the vector $\vec{B}$ is

$$\vec{B}(K) = \sum_{I=LB}^{LT} YF(I)*(XF(I)**(K - 1)) \quad K = 1,...,M, \quad (30)$$

and the Kth element of the vector $\vec{C}$ is just the Kth polynomial coefficient. Upon inversion we find the solution for $\vec{C}$

$$\vec{C} = A^{-1} \cdot \vec{B} \quad (31)$$

Both of the interpolator subroutines take due account of the end points of the arrays and minimize the number of calculations when more than one interpolated point falls between the same time values.

2.2 Digital Filtering

This section briefly reviews the theory of digital filters. Frequently in EMP work, the signal is contaminated by high-frequency noise arising both from the nature of the measurements and the digitization process which reduces the continuous signal to a digital record. This high-frequency component can generally be eliminated by passing the digital record through a low-pass digital filter. Another useful application of low-pass filters is in reducing the number of digital values needed to accurately calculate the Fourier transform of a waveform at low frequencies. For example, if there is a signal with frequency content up to 250 MHz, the Nyquist criterion is satisfied by sampling the signal every 2.0 nsec. If the signal has a 2-μsec duration, then 1000 numbers must be stored for the Fourier transform routine. However, if most of the significant frequency content is contained in a frequency band up to say 50 MHz, then after filtering, only 200 numbers must be stored to adequately represent the signal and obtain the Fourier transform without worrying about foldover effects. Bandpass and high-pass digital filters have application in EMP work when one is interested in studying only a certain region of the frequency spectrum so that correlations between equipment upset and damage and the induced signal can be determined.

2.2.1 Theory of Digital Filtering and Z Transforms

In EMP work, there is generally (after interpolation or as a result of the digitization process) a sequence of numbers $u(k)$, $k=0,...N$ that must pass through a linear discrete system (generally a difference equation) in order to limit, in some manner, the frequency content of the signal. The output of the linear system is denoted here by $y(k)$, $k=0,...,N$. By a linear discrete system is meant a system in which the output $y(k)$ is expressed as a linear combination of inputs and past outputs; thus,

$$y(k) + \sum_{j=1}^{n} a(j)y(k-j) = \sum_{\ell=0}^{m} b(\ell)u(k-\ell) , \tag{32}$$

and $y(k)$, $u(k) = 0$, $k < 0$.

The Z transform is used to simplify the analysis and synthesis of the digital filter represented by equation (32)--for example, generate the set of constants $a(j)$ and $b(\ell)$.

The Z transform of a sequence of numbers $f(k)$, $k=0,\ldots N$ $f(k) = 0$, $k < 0$ is defined by

$$Z[f(k)] = F(z) = \sum_{k=0}^{\infty} f(k) z^{-k} \quad , \tag{33}$$

where z is an arbitrary complex number.

The Z transform of the input-output signals are related to one another by

$$Y(z) = H(z) \, U(z) \quad , \tag{34}$$

where $U(z)$ and $Y(z)$ denote the Z transforms of the input and output signals respectively, and $H(z)$ is the system transfer function, and is given by

$$H(z) = \sum_{\ell=0}^{m} b(\ell) z^{\ell} / \left(1 + \sum_{j=1}^{n} a(j) z^{-j} \right) \tag{35}$$

Proof: Multiply each side of equation (32) by z^{-k} and sum over all k.

$$\sum_{k=0}^{\infty} y(k) z^{-k} + \sum_{j=1}^{\infty} a(j) \sum_{k=0}^{\infty} y(k-j) z^{-k} = \sum_{\ell=0}^{m} b(\ell) \sum_{k=0}^{\infty} u(k-\ell) z^{-k} \quad . \tag{36}$$

Using the properties of the Z transform, this can be rewritten as

$$Y(z) + \sum_{j=1}^{n} a(j) z^{-j} \, Y(z) = \sum_{\ell=0}^{m} b(\ell) z^{-\ell} \, U(z) \quad ; \tag{37}$$

or, as

$$Y(z) \left[1 + \sum_{j=1}^{n} a(j)z^{-j} \right] = \left[\sum_{\ell=0}^{m} b(\ell)z^{-\ell} \right] U(z) \quad . \tag{38}$$

Thus, equation (35) follows and may be rewritten as

$$H(z) = b \prod_{i=1}^{r} (z-z_i) / \prod_{i=1}^{n} \left(z-p_i\right) \quad . \tag{39}$$

Here, z_i and p_i are called the zeros and poles of the system transfer function $H(z)$.

One of the most important properties of the transfer function is the fact that the location of the zeros and poles has an enormous effect on how the system transmits different types of inputs. Thus, a system can be synthesized that will pass some inputs and reject others by a proper selection of the zeros and poles.

Assume the application of a sinusoidal input $u(k) = \sin(kwT)$ $k=0$, 1, 2,... to our systems. Then the resultant steady-state response, $y(k)$, is given by

$$y(k) = \left| H\left(e^{iwT}\right) \right| \sin(kwT+\theta) \quad . \tag{40}$$

Here, the sinusoidal response of the system is obtained by evaluating the system transfer function $H(z)$ at $z = e^{iwT}$, where w is the radian frequency of the input sinusoid and T the underlying sampling period. θ is the phase angle of $H\left(e^{iwT}\right)$--that is, $H\left(e^{iwT}\right) = \left| H\left(e^{iwT}\right) \right| e^{i\theta}$. To prove equation (40) let $u(k) = \sin(kwT)$ $k = 0$, 1, 2, ...; then, $U(z)$ is given by

$$U(z) = \frac{z\sin wT}{\left(z-e^{iwT}\right)\left(z-e^{-iwT}\right)} \tag{41}$$

The Z transform of the system response is given by,

$$Y(z) = \frac{H(z)\,z\sin wT}{\left(z-e^{iwT}\right)\left(z-e^{-iwT}\right)} \quad . \tag{42}$$

Since only stable systems are considered, all the poles of $H(z)$ must be inside the unit circle. Therefore, none of the poles of $H(z)$ is at e^{iwT} or e^{-iwT}. A partial fraction expansion of equation (42) yields

$$Y(z) = \frac{az}{\left(z-e^{iwT}\right)} + \frac{bz}{\left(z-e^{-iwT}\right)} + \text{terms due to poles of } H(z), \tag{43}$$

where

$$a = H\left(e^{iwT}\right)/2i \text{ and } b = -H\left(e^{-iwT}\right)/2i \quad .$$

Noting that $H\left(e^{iwT}\right) = H\left(e^{-iwT}\right)*$ and writing $H\left(e^{iwT}\right) = Me^{i\theta}$, we find that

$$Y(z) = \frac{M}{2}\left[\frac{ze^{i\theta}}{\left(z-e^{iwT}\right)} \quad \frac{-ze^{-i\theta}}{\left(z-e^{-iwT}\right)}\right] + \text{ terms due to poles of } H(z) \quad . \tag{44}$$

Taking the inverse Z transform of equation (44) we obtain

$$y(k) = \frac{M}{2i}\left[e^{i(kwT+\theta)} - e^{-i(kwT+\theta)}\right] + \text{transient response} \tag{45}$$
$$\text{generated by poles}$$
$$\text{of } H(z);$$

or,

$$y(k) = M\sin(kwT+\theta) + y_{transient}(k) \quad . \tag{46}$$

21

In the steady-state y transient (k) → 0 as k becomes large. Thus

$$y(k) = \left| H\left(e^{iwT}\right) \right| \sin(kwT+\theta) \quad . \tag{47}$$

The quantity $M=\left| H\left(e^{iwT}\right) \right|$ is called the system gain factor. To filter out a given sinusoid, pick a system so that M∿0; or, to amplify a given sinusoid, pick M>1.

As has been observed in equation (39), the transfer function is a ratio of polynomials in the variable z. Therefore, the system gain factor that is equal to the magnitude of the transfer function evaluated at z=exp(iwT) may always be expressed as a ratio of polynomials in the variables cos(wT) and sin(wT). Thus, different filters can be synthesized by investigating ratios of trigonometric functions. For example, a low-pass filter with half-power point w, has the following squared gain factor,

$$\left| H\left(e^{iwT}\right) \right|^2 = \cfrac{1}{1 + \cfrac{\tan^{2n}(wT/2)}{\tan^{2n}\left(w_1 T/2\right)}} \quad . \tag{48}$$

By a considerable amount of algebraic manipulation equation (48) can be written as

$$\left| H(z) \right|^2 = \frac{\tan^{2n}\left(w_1 T/2\right)\left(1+z\right)^{2n}}{\left[\tan^{2n}\left(w_1 T/2\right) + \left(-1\right)^n\right]\left[(z-p_1)\left(z-p_2\right)\cdots\left(z-p_{2n}\right)\right]} \quad , \tag{49}$$

where z = exp(iwT), and the 2n poles p_i are given by

$$p_i = \frac{1 - \tan^2\left(w_1 T/2\right) + \sqrt{-1}\ 2\tan\left(w_1 T/2\right)\sin\theta_i}{1 - 2\tan\left(w_1 T/2\right)\cos\theta_i + \tan^2\left(w_1 T/2\right) r} \quad , \tag{50}$$

where,

$$\theta_i = (i-1) \, \pi/2, \text{ n odd}$$

$$= (2i-1) \, \pi/2, \text{ n even} \, .$$

It can be shown that of the 2n poles, p_i, exactly n lie inside the unit circle and n outside. Let p_1, p_2,...,p_n denote the n poles inside the unit circle. The transfer function that has the desired squared-gain factor is given by

$$H(z) = \frac{b(1+z)^n}{\left(z-p_1\right)\left(z-p_2\right) \cdots \left(z-p_n\right)} \, , \qquad (51)$$

where b is chosen so that the steady-state, unit-step response has magnitude one, $H(1)=1$; thus,

$$b = \frac{\left(1-p_1\right)\left(1-p_2\right)\cdots\left(1-p_n\right)}{2^n} \, . \qquad (52)$$

The remaining poles p_{n+1}, p_{n+2},...,p_{2n} associated with $|H(z)|^2$ can be shown to arise from the process of determining the squared-gain factor.

From the foregoing, a well-defined procedure exists for synthesizing a low-pass filter. It can be summarized in the following steps:

(a)   Determine the half-power point w_1,

(b)   Determine the value n--using equation (17)--by specifying the gain at frequency w_2,

(c)   Find the n roots p given by equation (49) which satisfy $|p_i| < 1$, and

(d)   Determine the difference equation which has the transfer function given by equations (51) and (52).

A squared-gain factor that corresponds to that of a high-pass filter is given by

$$\left| H\left(e^{iwT}\right) \right| \;=\; \cfrac{1}{1 \;+\; \cfrac{\cot^{2n}(wT/2)}{\cot^{2n}\left(w_2 T/2\right)}} \tag{53}$$

Here, the half-power point is denoted by w_2. Formulas analagous to equations (50) and (51) may be derived Then, if the poles and zeros of a low-pass filter with half-power point $(\pi/T - w_2)$ are rotated through π radians in the complex plane, the pole-zero pattern of a high-pass filter is obtained with half-power point w_2 and the same gain-factor falloff outside its passband is obtained. Thus, if we have $H(z)$ for the low-pass filter given by

$$H(z) \;=\; b\left(1+z\right)^{n} \Big/ \prod_{i=1}^{n} \left(z - p_i\right) \quad, \tag{54}$$

then the high-pass filter is given by $H'(z)$

$$H'(z) \;=\; b'\left(z-1\right)^{n} \Big/ \prod_{i=1}^{n} \left(z + p_i\right) \quad. \tag{55}$$

2.2.2   Examples of Filtering EMP Data

To illustrate the implementation of the algorithms described in section 1, a typical EMP waveform was selected from the vast amount of data collected at the Polk City AUTOVON EMP tests and processed. Figure 4 plots the digitized waveform after it has been digitized, time ordered, and interpolated at 1.63-nsec intervals using a Lagrange interpolator.

Figure 5 plots the power spectrum after passing the digital record through a fast Fourier transform routine. All the power is contained in two peaks at 13.2 and 24.0 MHz. Several filters were then synthesized and the digital record passed through filters before processing it through the fast Fourier transform routines.   Table I lists the half-power points, w_1, the gain at the higher frequency w_2 used to determine n, and the filter coefficients $a(j)$, $j=1,\ldots,n$, $b(j)$, $j=0\ldots,n$, which were calculated for each synthesized filter. Figures 6 through 12 are plots of the power spectra obtained by using a fast Fourier transform routine on the filtered-time series. The general result is fairly evident from these plots--namely, that as the number of poles is increased and consequently the gain rolloff at the half-power point is increased, a sharper filtering is obtained.

Although it is not apparent from the plots of the power spectra presented, there is much greater definition of the peaks in the power spectrum relative to the noise background. The high-frequency noise was reduced by a factor of 100.

24
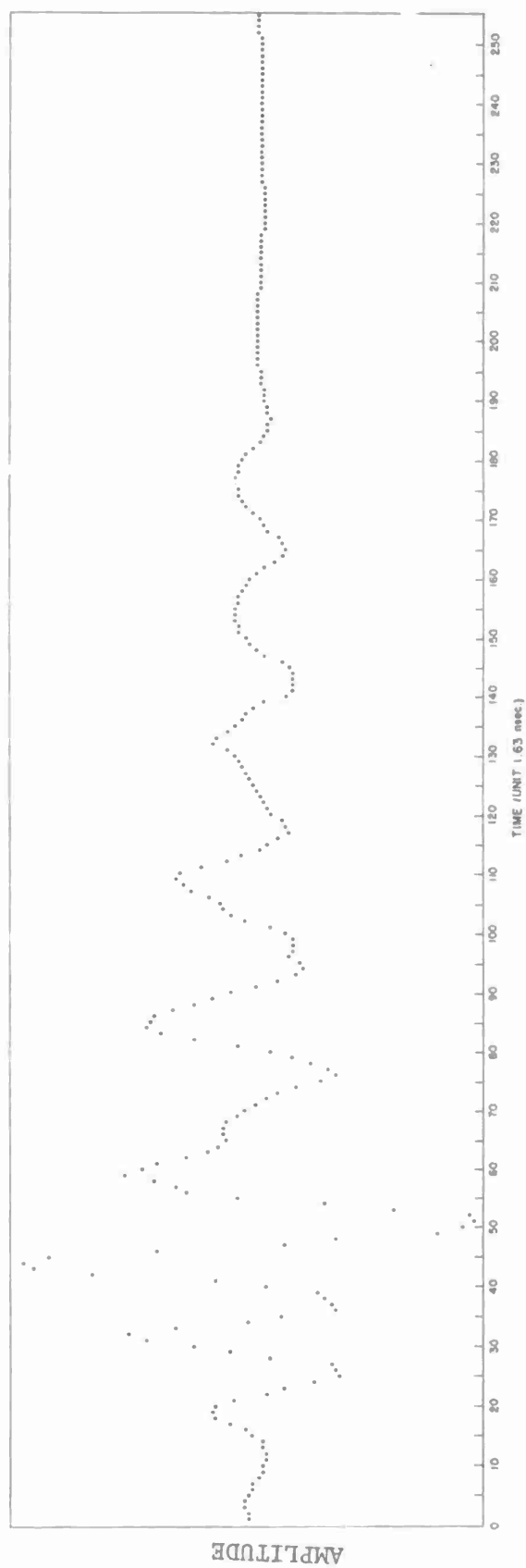
Figure 4. Plot of equispaced digital times series.

AMPLITUDE

TIME (UNIT 1.63 msec.)
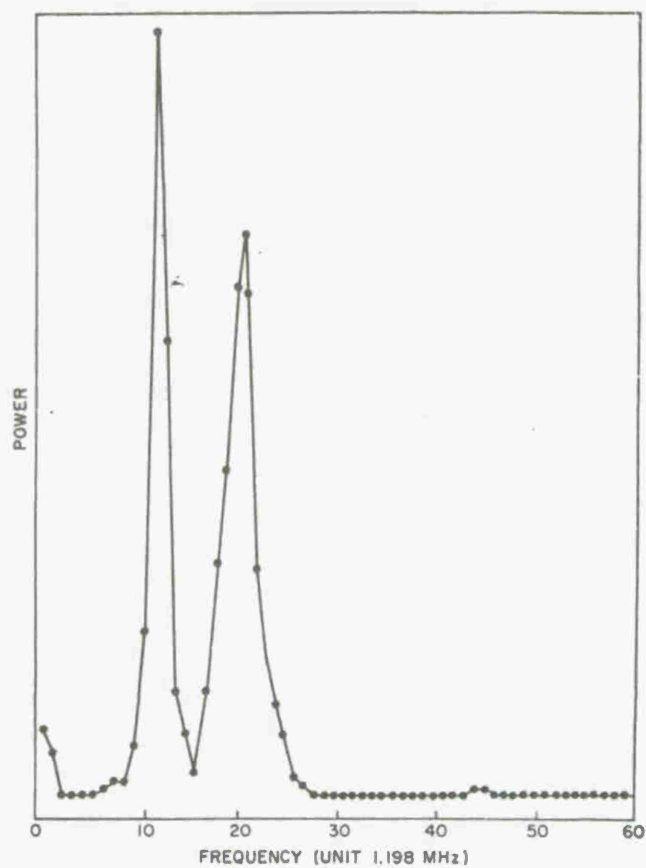
Figure 5. Plot of power spectrum obtained without filtering.

TABLE I. PARAMETERS CALCULATED FOR EACH SYNTHESIZED FILTER

Filter No.	Half-power point W_1 (MHz)	Gain $W = W_2 \gg W_1$	W_2	Sampling Interval T (μsec)	n	$a(j), j=1,\ldots,n$	$b(j), j=1,\ldots n+1$
1	20	0.4	25	1.63	1	-0.8135657E00	+0.9321713E00
							+0.9321713E00
2	20	0.2	25	1.63	4	-0.3465070E 01	0.8540331E-04
						0.4533676E 01	0.3416131E-03
						-0.2652045E 01	0.5124197E-03
						0.5848047E 00	0.8540331E-04
3	20	0.1	25	1.63	5	-0.4337327E 01	0.8235322E-05
						0.7563597E 01	0.4117661E-04
						-0.6625226E 01	0.8235322E-04
						0.2913780E 01	0.8235322E-04
						-0.5145599E 00	0.4117661E-04
							0.8235322E-05
4	20	0.07	25	1.63	6	-0.5208673E 01	0.7937438E-06
						0.1135041E 02	0.4762463E-05
						-0.1324103E 02	0.1190616E-04
						0.8718896E 01	0.1587487E-04
						-0.3071778E 01	0.1190616E-04
						0.4523243E 00	0.4762463E-05
							0.7937438E-06
5	20	0.05	25	6.52	6	-0.2848268E 01	0.1299385E-02
						0.3845003E 01	0.7796306E-02
						-0.2959958E 01	0.1949077E-01
						0.1351626E 01	0.2598769E-01
						-0.3426920E 00	0.1949077E-01
						0.3745000E-01	0.7796306E-02
							0.1299385E-02
6	18	0.2	25	1.63	3	-0.2631725E 01	0.6571114E-03
						0.2328194E 01	0.1971334E-02
						0.6912119E 00	0.1971334E-02
							0.6571114E-03
7	17.5	0.05	25	1.63	5	-0.4420094E 01	0.4385640E-05
						0.7844978E 01	0.2192819E-04
						-0.6986225E 01	0.4385639E-04
						0.3120767E 01	0.4385639E-04
						-0.5592871E 00	0.2192819E-04
							0.4385640E-05
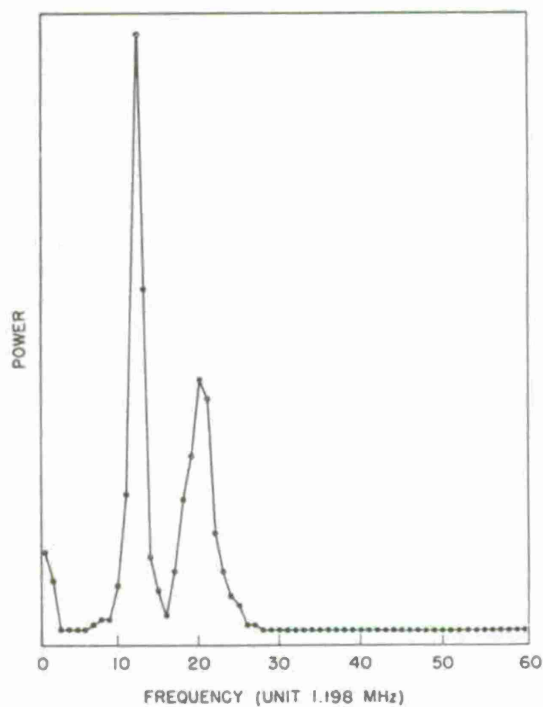
Figure 6. Plot of power spectrum obtained after passing the digital record through filter No. 1 (table I).

Figure 7. Plot of power spectrum obtained after passing the digital record through filter No. 2 (table I).
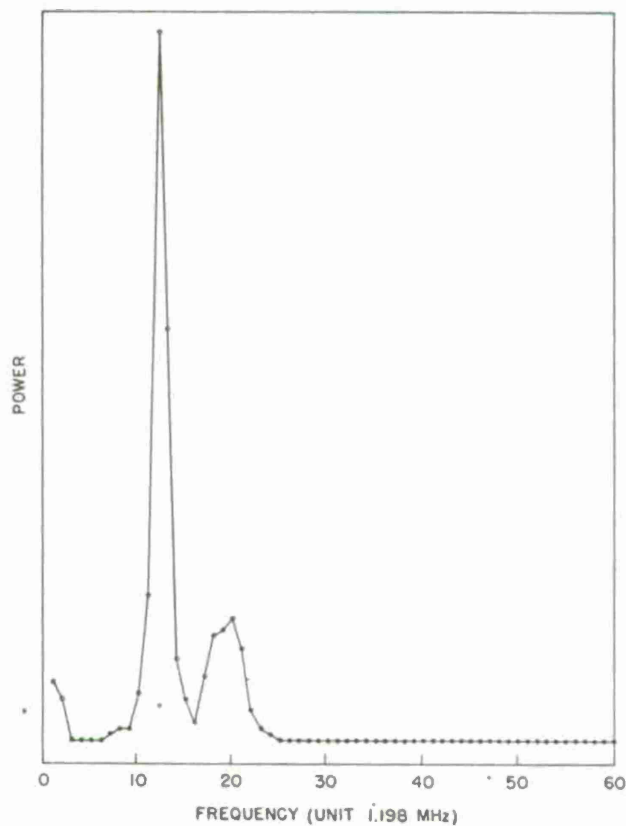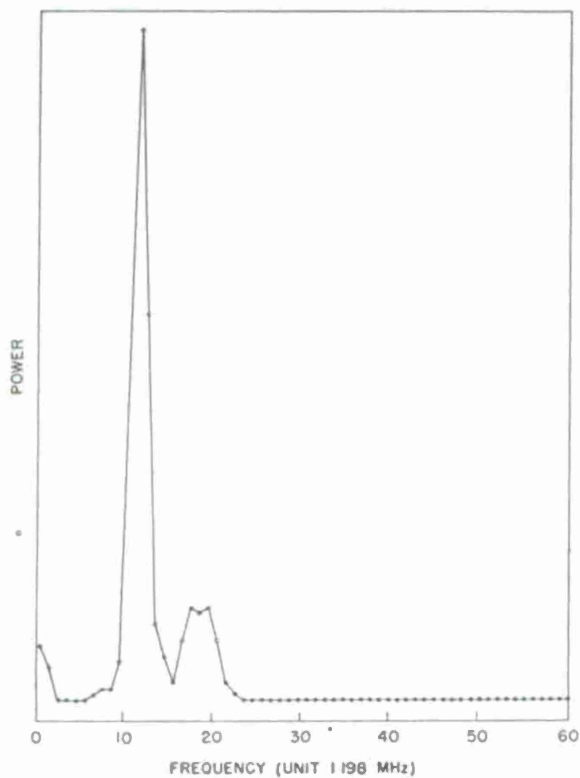
Figure 8.   Plot of power spectrum obtained
after passing the digital
record through filter No. 3
(table I).

Figure 9.   Plot of power spectrum
obtained after passing
the digital record
through filter No. 4
(table I).

29
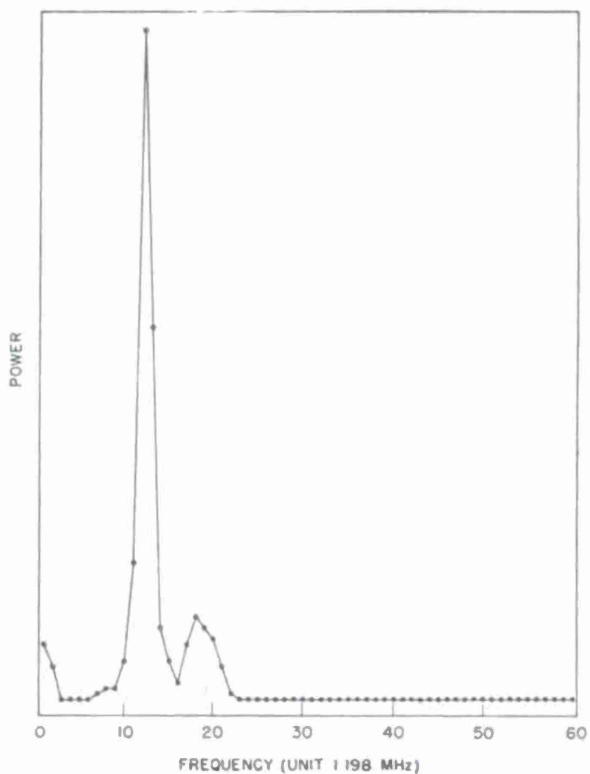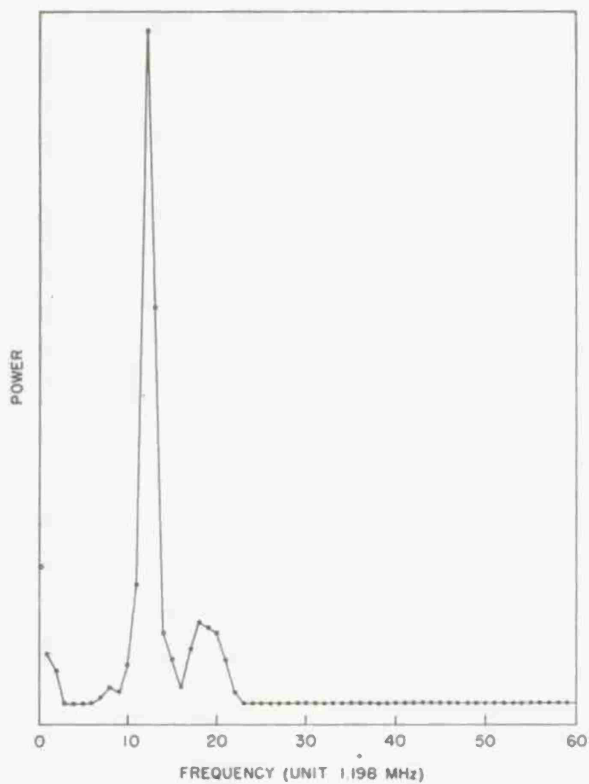
Figure 10. Plot of power spectrum obtaine after passing the digital record through filter No. 5 (table I).

Figure 11. Plot of power spectrum obtained after passing the digital record through filter No. 6 (table I).
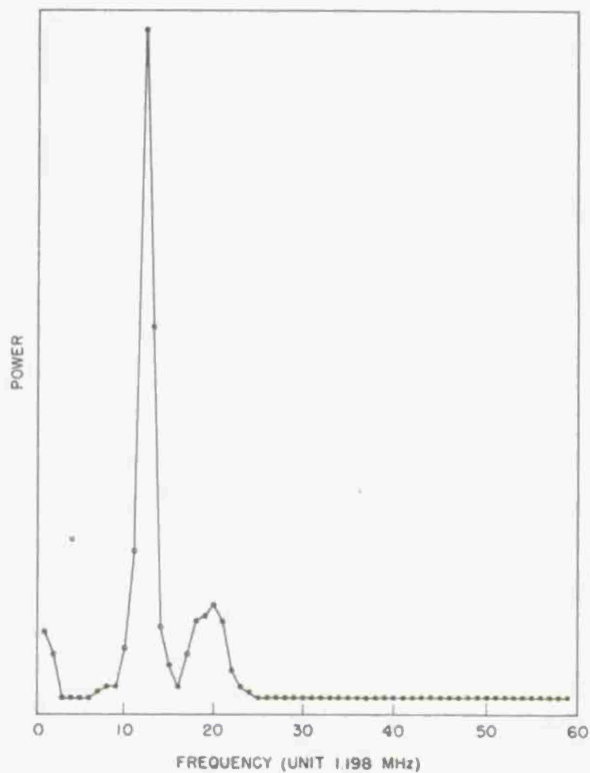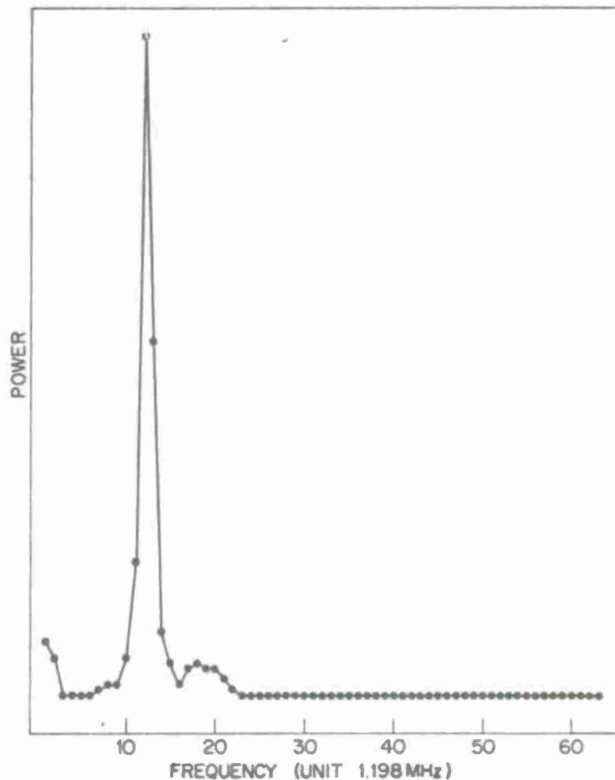
Figure 12.   Plot of power spectrum obtained after passing the digital
record through filter No. 7 (table I).

2.3   Transform Techniques

Three different popular transform techniques are discussed below.

2.3.1   Fast Fourier Transform

The fast Fourier transform is by far the method most preferred for
generating the Fourier transform of a digital-time series.   A short
discussion of the method is given in this section.

$F(\omega)$ of a function $f(t)$ is defined as

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} \, dt \quad . \tag{56}$$

If $f(t)$ is nonzero only over a finite time interval T, then it is a good
approximation to write the Fourier transform as a discrete sum,

31

$$F(\omega) = \Delta t \sum_{k=o.}^{N-1} f(k\Delta t) e^{-\omega k \Delta t}, \ N\Delta t = T \ . \tag{57}$$

The inverse transform is

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) \ e^{i\omega t} \ d\omega \ . \tag{58}$$

If $F(\omega)$ has no (or a negligible) frequency content above ω_{max} the inverse transform can be written as

$$f(t) = \frac{\Delta\omega}{2\pi} \sum_{r=o}^{N-1} F(r\Delta\omega) e^{itr\Delta\omega}, \ N\Delta\omega = 2\omega_{max} \ . \tag{59}$$

For a band-limited signal, we know from the Nyquist criterion that the sampling interval Δt must be chosen so that $\Delta t = 2\pi/2\omega_{max}$. Similarly, for a time-bounded signal, the frequency-sampling interval must be chosen so that $\Delta\omega = 2\omega/T$,

$$\Delta t \Delta\omega \leq \frac{T}{N} \frac{2\pi}{T} = \frac{2\pi}{N} \ . \tag{60}$$

Thus,

$$F(r\Delta\omega) = \Delta t \sum_{k=o}^{N-1} f(k\Delta t) e^{-2\pi i r k/N} \ , \ r=o, \ 1,\ldots,N - 1 \ . \tag{61}$$

For notational convenience, we write $F_r = F(r\Delta\omega)$, $f_k = f(k\Delta t)$, $W = e^{-2\pi i/N}$, and drop the factor Δt.

$$F_r = \sum_{k=o}^{N-1} f_k \ W^{rk} \qquad r = o,\ldots, \ N - 1 \ . \tag{62}$$

Divide the time series of points, f_k, into two functions g_k and h_k,

$$g_k = f_{2k}, \qquad\qquad k = 0, 1, 2 \ldots \frac{N}{2} - 1 \qquad . \qquad (63)$$
$$h_k = f_{2k+1},$$

The discrete Fourier transforms of g_k and h_k are G_r and H_r, respectively.

$$G_r = \sum_{k=0}^{(N/2)-1} g_k \, e^{-2\pi i r k/(N/2)} \quad,$$

$$r = 0, 1, \ldots \frac{N}{2} - 1 \qquad (64)$$

$$H_r = \sum_{k=0}^{(N/2)-1} h_k e^{-2\pi i r k/(N/2)} .$$

Equations (61) and (62) may be rewritten as

$$F_r = \sum_{k=0}^{(N/2)-1} f_{2k} e^{-2\pi i r k/(N/2)} + e^{-2\pi i r/N} \sum_{k=0}^{(N/2)-1} f_{2k+1} e^{-2\pi i r k/(N/2)} \quad .(65)$$

Using equations (63) and (64), we have

$$F_r = G_r + W H_r \qquad\qquad 0 \le r < N/2$$

$$F_{r+N/2} = G_r - W H_r \qquad 0 \le r < N/2 \quad . \tag{66}$$

We may therefore compute the Fourier transform of a function sampled N times by evaluating two Fourier transforms of the function sampled N/2 times. The computations of G_r and H_r can be reduced to the evaluation of sequences of N/4 samples. If $N = 2^n$, n such reductions can be made by applying equations (63) and (66) first for N, then N/2, and finally for a two-point function. The Fourier transform of a one-point function

is just the sample itself. Beside the obvious savings in computer
running time by using this successive reduction scheme, the transform
can also be done in place--that is, in each stage of reduction the
intermediate results are written over the original array. This
in-place reduction requires a rearrangement of the original array called
bit reversal (subroutine SRTFUR). A complete rearrangement and
overwriting sequence for an eight-point sampled function is illustrated
in figure 13.
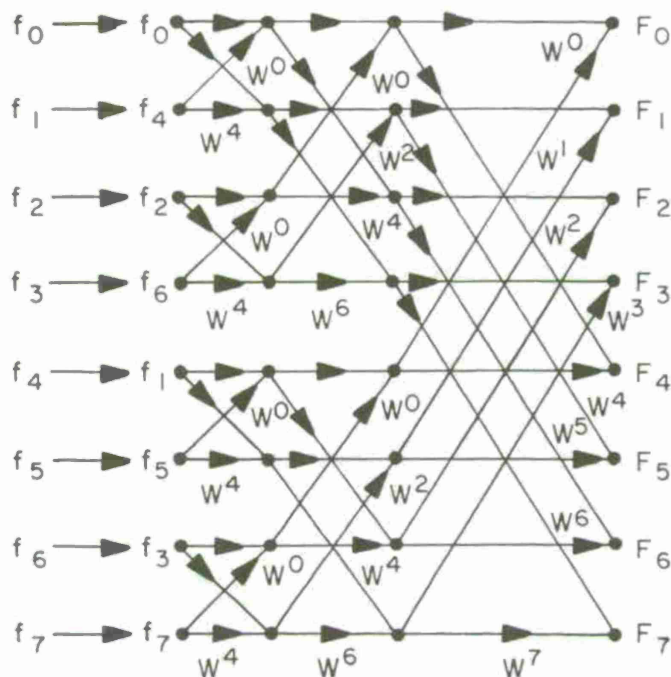
Figure 13.   Rearrangement and overwriting sequence for an eight-point
sampled function.

Each arrow in the diagram means that the term at the origin of the arrow
must be added. A variable next to the arrow acts as a multiplier to the
additive term. Thus, at the first overwriting, sequence f_0 is replaced
by $f_0 + W^0 * f_4$ and f_4 is replaced by $f_0 + W^4 * f_4$.

Subroutine FFT will compute the transform of any array with 2^n elements
as described above. The only restriction is that imposed by the finite
memory size of the computer being used. Several examples of the power
spectrum derived from the real and imaginary parts of the transform
generated by FFT are shown in figures 5 through 12.

2.3.2 Fast Walsh Transform

The previous section transforms the digital-time series to the frequency domain by using a particular set of orthogonal functions--namely sines and cosines. Another set of orthogonal functions, which are used extensively in communications theory, are the Walsh functions that are used primarily to represent logic signals. Their most appealing feature is that the digital Walsh transform algorithm is about an order of magnitude faster than the Fourier transform algorithm.

The Walsh functions are wal(k,θ), sal(k,θ), and cal(k,θ),

$$wal(2k,\theta) = cal(k,\theta)$$

$$wal(2k-1,\theta) = sal(k,\theta)$$

They are defined on the time interval T, θ is the normalized time $\theta = t/T$, and k is called the sequency. The sequency is equal to the average number of zero crossings of the function per unit time. The functions sal and cal are similar to the sine and cosine functions. The sequency of the Walsh functions plays a similar role as the frequency for the sinusoidal functions. One definition of the Walsh functions is through a difference equation,

$$wal(2k+p,\theta) = (-)^{[k/2] + p} \left[wal\left[k, 2\left(\theta, \frac{1}{4}\right)\right] + (-)^{k+p} wal\left[k, 2\left(\theta - \frac{1}{4}\right)\right] \right], \quad (67)$$

where

$k = 0, 1, 2, \ldots,$

$[k/2]$ is the largest integer less than or equal to $k/2$,

$p = 0$ or 1, and

$$wal(0,\theta) = \begin{cases} 1 & \theta \leq \frac{1}{2} \\ 0 & \theta > \frac{1}{2} \end{cases} \quad (68)$$

A few of the Walsh functions are shown in figure 14.
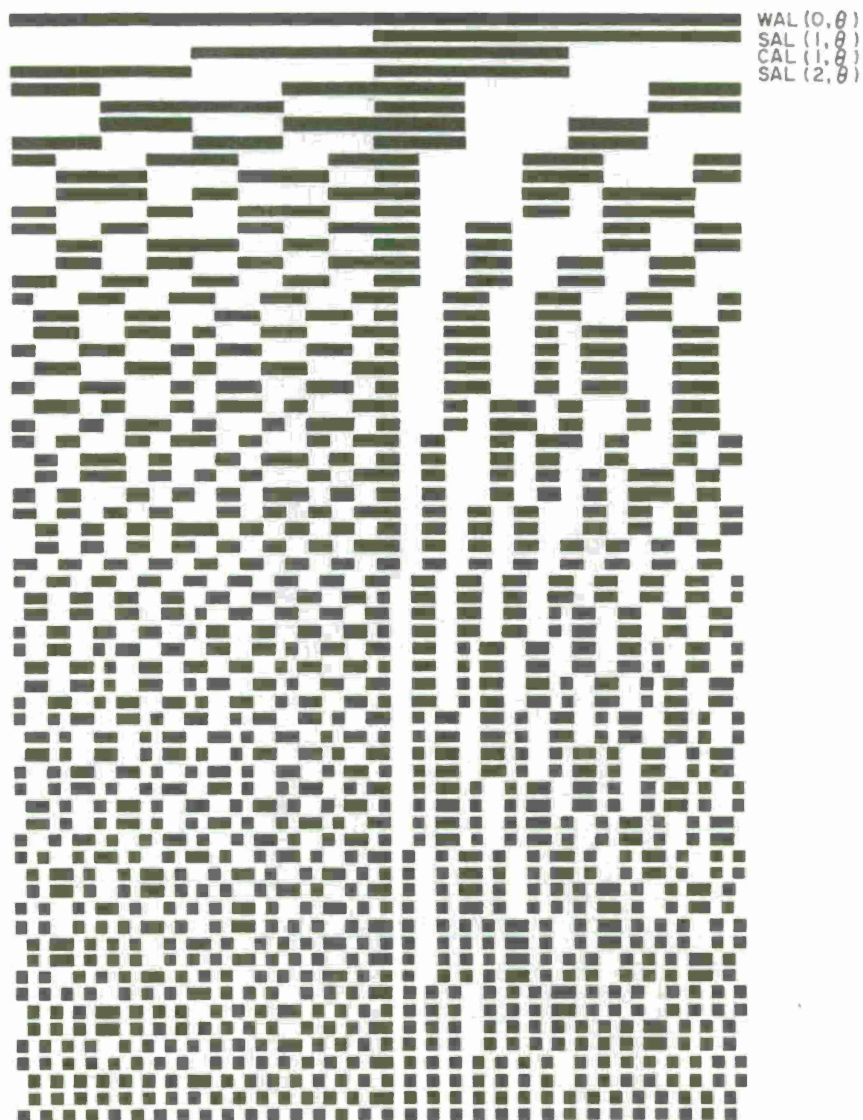
Figure 14. Examples of Walsh functions for interval $-\frac{1}{2} < \theta < \frac{1}{2}$ (dark areas imply +1, light areas imply -1).

A signal f(t) may be expanded in a Walsh series

$$f(t) = a(o)wal(o,t) + \sum_{k=1}^{\infty} \left[a_c(k)cal(k,t) + a_s(k)sal(k,t) \right] ,$$

$$a(o) = \int_{-T/2}^{T/2} f(t)dt ,$$

$$a_c(k) = \int_{-T/2}^{T/2} f(t)cal(k,t) dt ,$$

$$a_s(k) = \int_{-T/2}^{T/2} f(t)sal(k,t)dt . \tag{69}$$

Just as in a Fourier expansion, the sum of the squares of the expansion coefficients give the sequency energy spectrum. The Walsh power is

$$E(k) = a_c^2(k) + a_s^2(k) . \tag{70}$$

To evaluate the coefficients, a fast Walsh transform algorithm can be derived that is similar to the fast Fourier technique. The main difference is that the reduction cannot be done in place. The steps for an eight-point-sampled function is shown in figure 15. The arrows have the same meaning as in the overwriting sequence for the FFT. Thus, at the first overwrite, f_0 is replaced by f_0+f_1.

For the EMP data of figure 4, the Walsh power was calculated and the results plotted in figure 16. The same figure plots the Walsh power after filtering the data through filter No. 7, table I.
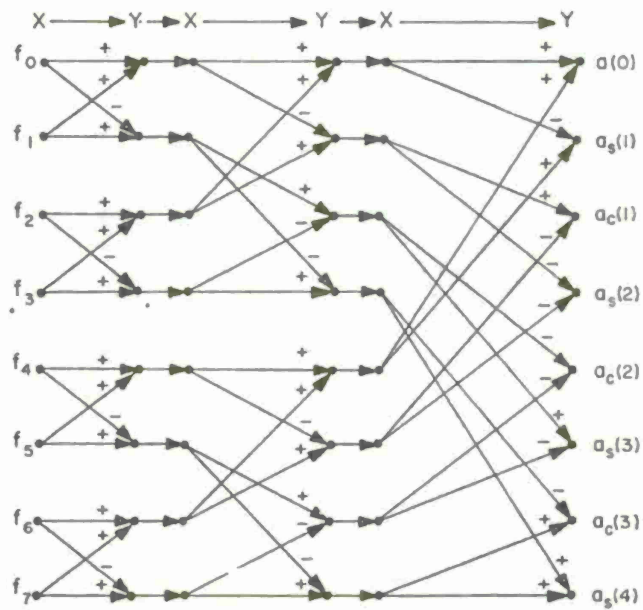
Figure 15.   Overwriting sequence for Walsh transform.
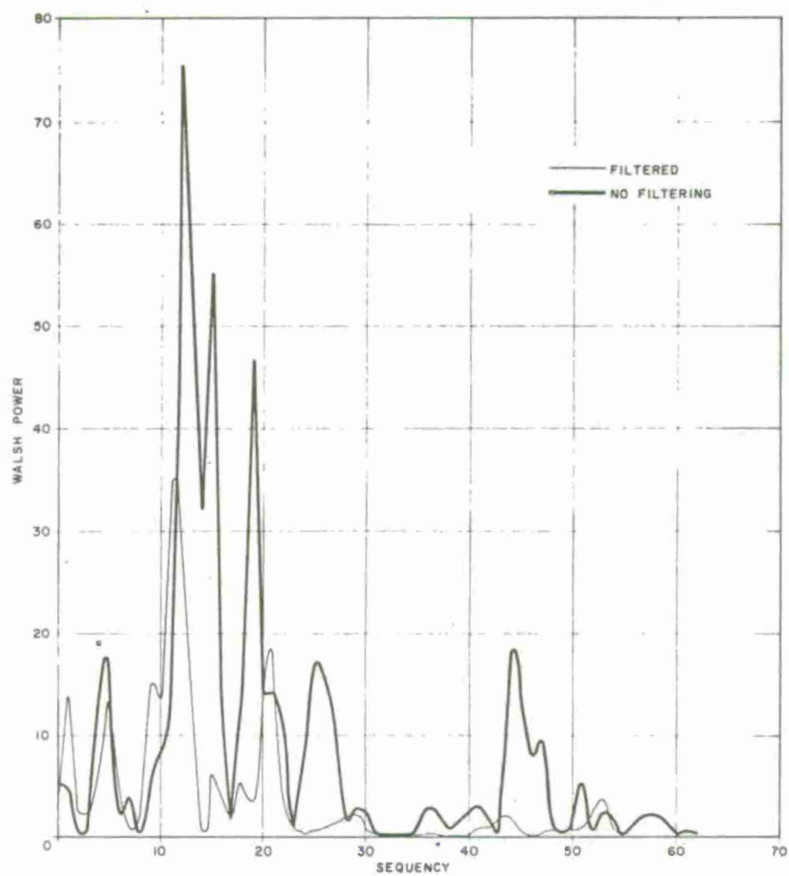
Figure 16.   Walsh power for the time series of figure 4.

38

2.3.3 Refined Spectral Densities and Autocorrelation Function

In EMP work, there is a great need for implementing numerical algorithms for the processing and storage of digitized waveforms, which are the main output of all EMP tests. This subsection describes one algorithm to generate the power spectrum of a digital record from its autocorrelation function. The theory presented below is illustrated by several examples.

A procedure will now be defined for estimating the power spectrum of a uniformly spaced, discrete time series of finite length.

If $C(\tau)$ is the autocovariance function for a time waveform $X(t)$, then by definition

$$C(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_{-T/2}^{T/2} X(t)X(t+\tau)dt \quad . \tag{71}$$

The power spectrum $P(w)$ of the time waveform is then given by

$$P(w) = 2 \int_{0}^{\infty} \cos(w\tau) \; C(t)dt \; . \tag{72}$$

For a uniformly spaced discrete-time series of finite length, denoted by X_0, $X_1, \ldots,$ X_n compute the mean lagged products, C_r, with lag interval $\Delta\tau = h\Delta t$, and Δt is the time interval between adjacent values of the time series.

$$C_r = \frac{1}{n-hr} \sum_{q=o}^{q=n-hr} X_q X_{qthr}, \; r=0, \; 1, \ldots, m, \; m \leq \frac{n}{h} \quad . \tag{73}$$

Next, compute the "raw spectral density estimates" Vr.

$$V_r = \Delta\tau \cdot \left[C_o + 2 \sum_{q=1}^{q=m-1} C_q \cos \frac{qr\pi}{m} + C_m \cos r\pi \right] \tag{74}$$

The frequency corresponding to r is $r/2m\Delta\tau$.

We next calculate the refined spectral density estimates according to

$$U_r = 0.23 \ V_{r-1} + 0.54 \ V_r + 0.23 \ V_{r+1} \quad . \tag{75}$$

To illustrate the implementation of the algorithms given by equations (73), (74), and (75), refer once again to figures 4 and 5. Figure 4 plots the digitized waveform after it has been digitized, time ordered, and interpolated at 1.63-nsec intervals with a Lagrange interpolator. Figure 5 plots the power spectrum after passing the digital record through a fast Fourier transform routine. Most of the power is contained in the two peaks at 13.2 and 24.0 MHz.

Figures 17 through 23 are plots of the mean lagged products and spectral densities for different values of the lag interval, Δt. It is seen that all of the frequency content of the power spectrum is accurately calculated until the lag interval exceeds the Nyquist sampling rate, $\Delta > t_N = 1/2f_{max}$, where f_{max} is the largest expected frequency content of the record. For these data, $\Delta t_N = 20$ nsec. Figure 22 shows the power spectrum clearly broadened and thus fold-over effects on the lower frequency peak. Since the autocorrelation function is close to zero for times greater than 0.5 µsec, the power spectrum can be accurately calculated with a lag of 1.0 and a lesser number of mean lagged products (fig. 23).
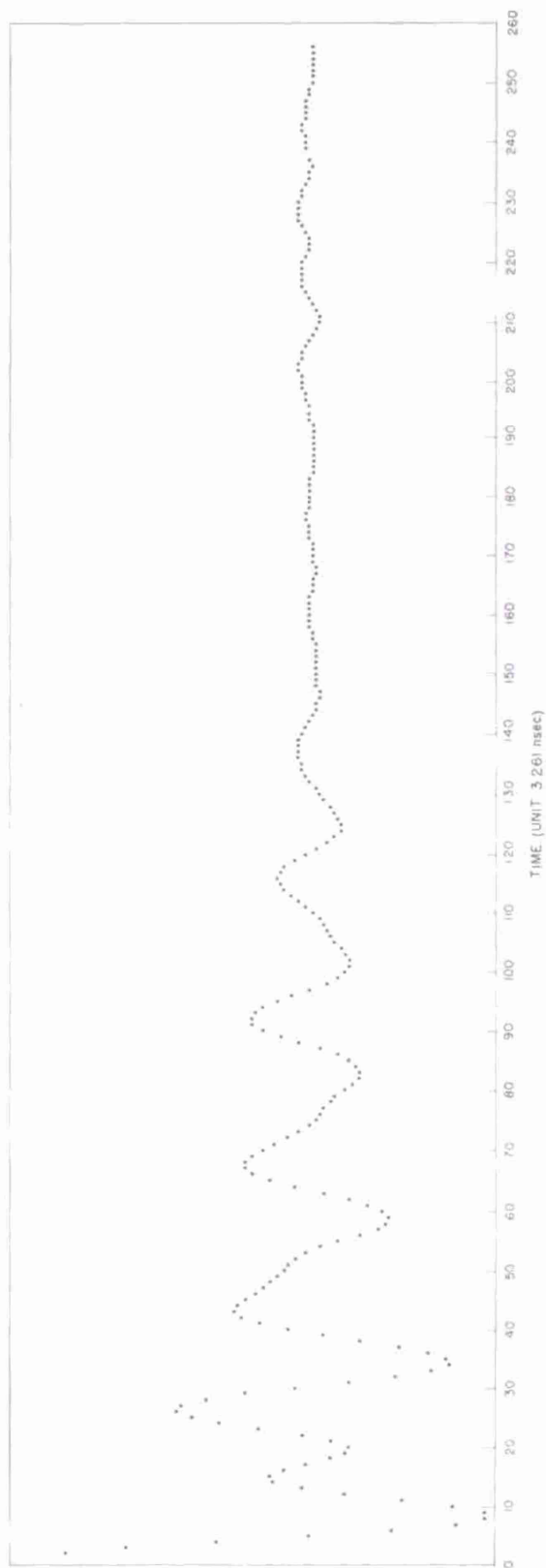
Figure 17. Plot of mean lagged products, C_r, $r=1,\ldots,256$. The log interval, $\Delta\tau$, is given by $\Delta\tau=\bar{H}\cdot\Delta\tau=3.61$ nsec.

TIME (UNIT 3.261 nsec)
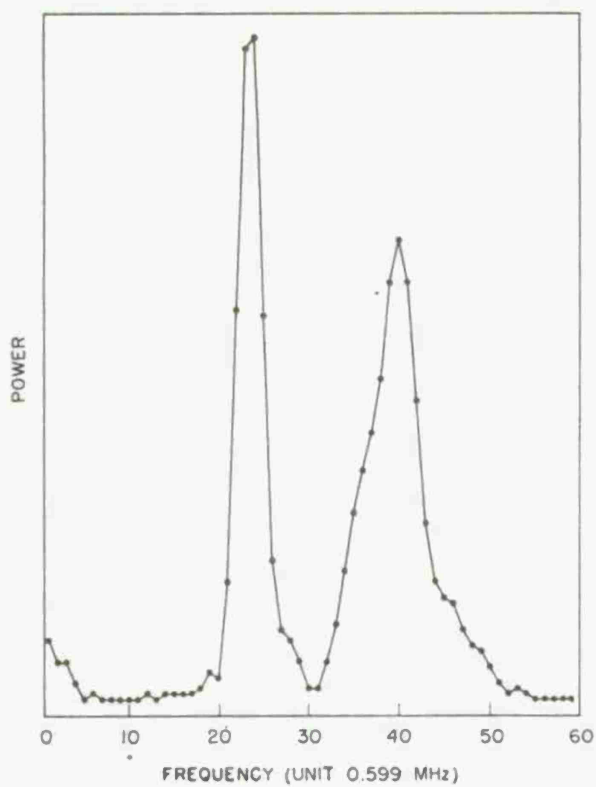
Figure 18. Refined spectral density estimates for the autocorrelation function of figure 17.

Figure 19. Plot of mean lagged products C_r, r=1, ...,64.
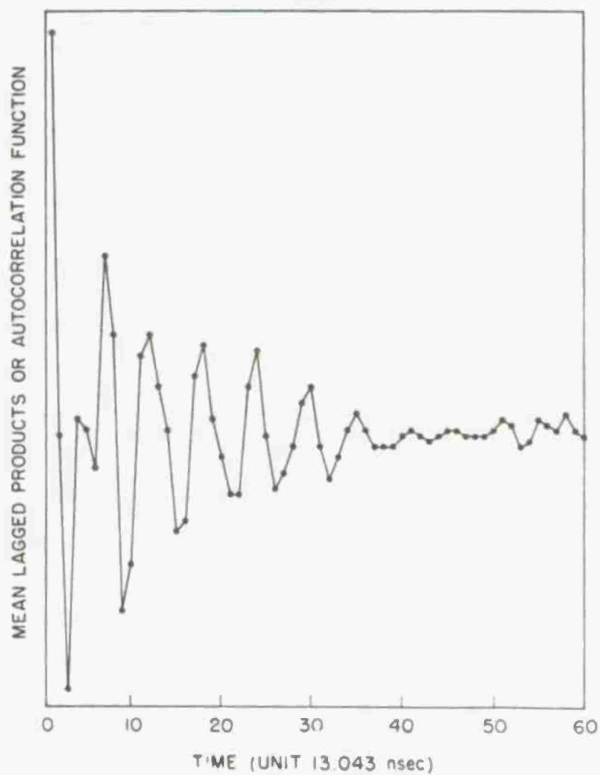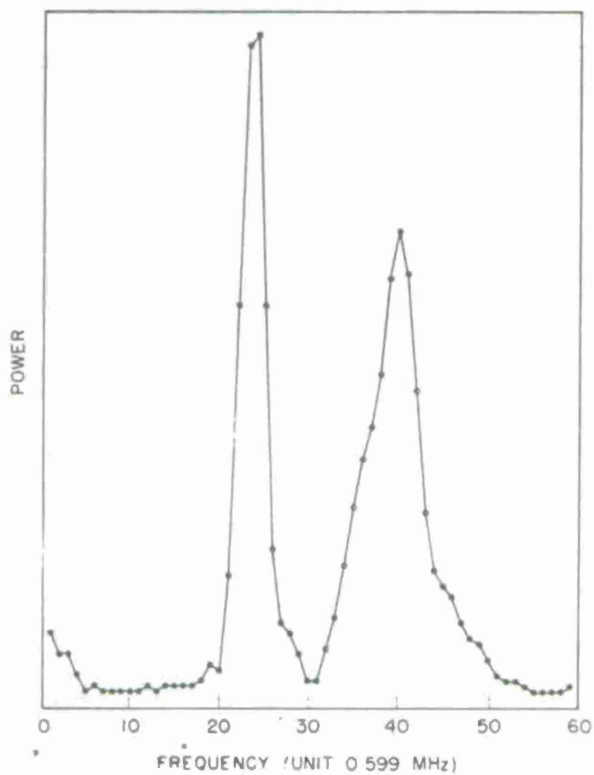
Figure 20.   Refined spectral density
estimates for the
autocorrelation func-
tion of figure 19.

Figure 21.   Plot of mean lagged
products, C_r, r=1
...,32.

43
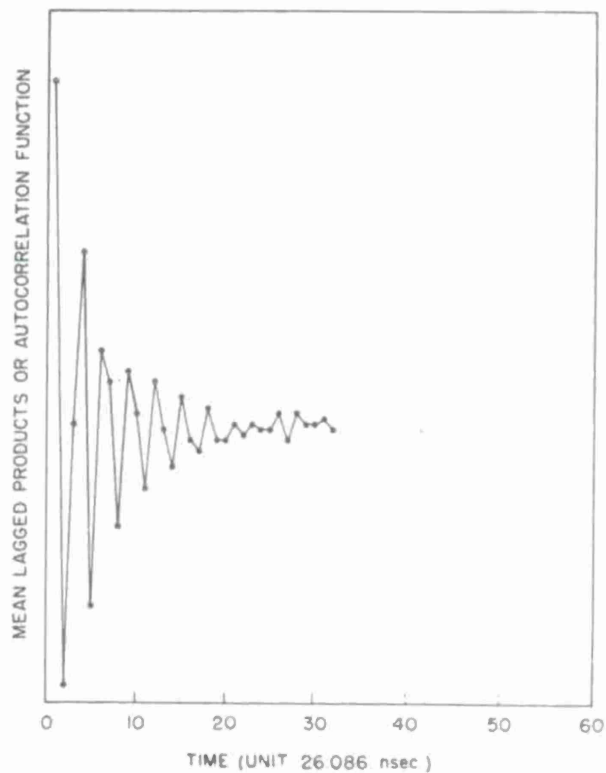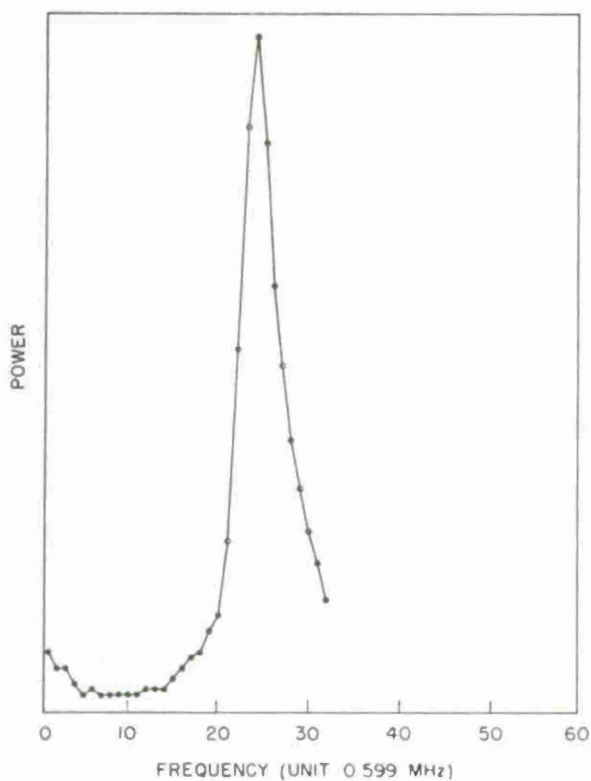
Figure 22. Refined spectral density estimates for the autocorrelation function of figure 21.

Figure 23. Refined spectral density estimated, obtained after truncation of the autocorrelation function of figure 6 at 0.1502 μsec.

44
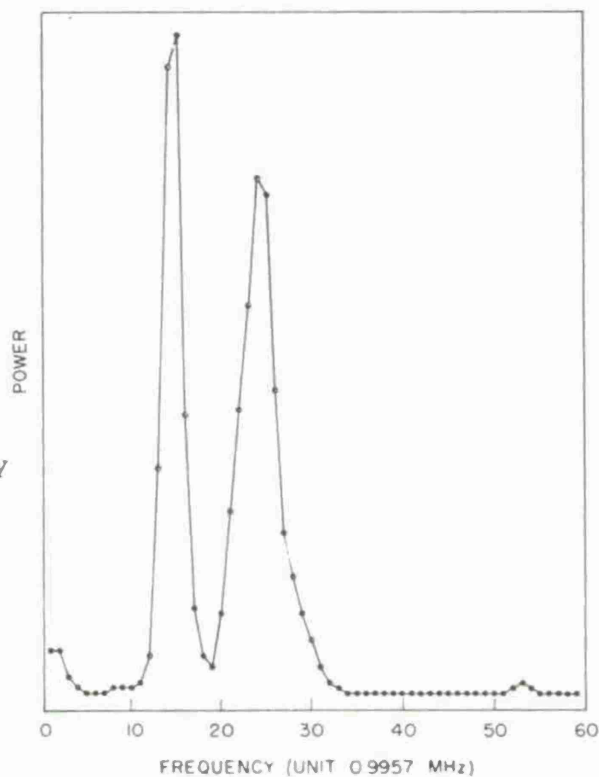
3. PROGRAM DESCRIPTION AND INPUT FORMATS

A flow chart of the main program is presented in figure 24. Each subroutine performs a definite signal analysis function. As indicated, the user is privileged to employ the technique of his choice by setting the control parameters on the input cards. The subroutines and their functions are listed.

3.1 Subroutine Descriptions

MAIN: This routine directly or indirectly calls the rest of the subroutines and thus controls the passage of the program through all the signal analysis options.

READIN: This routine reads in all input parameters and data.

SCARTP: This routine calculates the rotation angle, the sine and cosine of the rotation angle, the origin (center) of the scope graticule, and the scale factors (tablet units/div) along both the time and voltage axes.

ROT: This routine rotates and scales the input time and amplitude arrays given in digitization-tablet units into units of time and volts.

CSTOUT: This routine checks the time ordering of the digital time series. It discards those points from the array whose time values, t_i, $i = 1,\ldots,N$, do not satisfy $t_{i+1} \geq t_i$. On those points that have the same time values, the program averages the amplitude values.

LNYQ: This routine interpolates at the Nyquist sampling intervals with a linear interpolator. If the time value, X, satisfies $XF(I-1) < X < XF(I)$, where XF is the time array, then the interpolated amplitude is $Y = C1 * X + C2$ where

$$C1 = (YF(I)-YF(I-1))/(XF(I)-XF(I-1))$$
$$C2 = (YF(I-1)*XF(I)-YF(I)*XF(I-1))/(XF(I)-XF(I-1))$$

NYQST: This routine interpolates at the Nyquist sampling intervals by using a least-squares interpolator. If the time value, X, satisfies $XF(I-1) < X < XF(I)$ where XF is the time array, then a least-squares fit of the function $\Phi = a_1 X(I)+a_2$ is made to the set of points $(XF(J), YF(J))$, $J=I-2$, $I-1$, I, $I+1$. Thus, this routine is also a linear interpolator. Program modifications can be made to increase the number of points and/or the degree of the polynomial used in the fitting procedures.

45
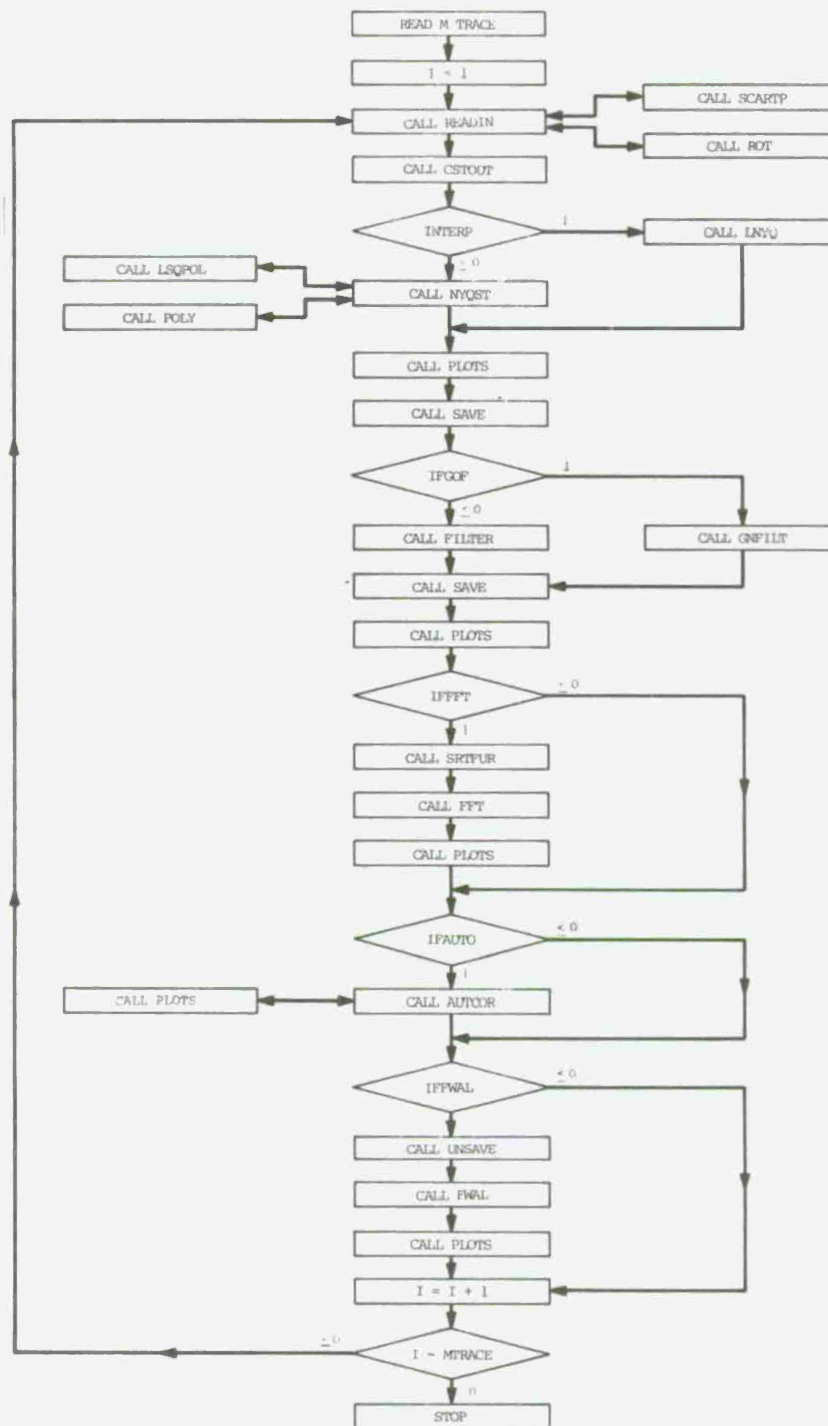
Figure 24. Flow chart for signal analysis program.

46

GNFILT: This routine will calculate the coefficients A(J) and B(J) for a Butterworth linear recursive low-pass digital filter. If Y(K) and U(K) are the output and input arrays to the filter, then

$$Y(K) = - \sum_{J=1}^{N} A(J)*Y(K-J) + \sum_{J=1}^{N+1} B(J)*U(K-J+1) \quad .$$

After calculating the coefficients, the routine filters the data.

FILTER: This routine filters the data through a Butterworth linear low-pass digital filter when the coefficients are known.

SRTFUR: This routine performs a *bit reversal* of the original Nyquist array. If I is the element of the array, then $I = \sum_{n=0}^{m} a(n)*(2**n)$, $a(n) = 0$ or 1. The element specified by I is exchanged with the element specified by J, where $J = \sum_{n=0}^{m} a(m-n)*(2**n)$.

FFT: This routine calculates the Fourier transform of a Nyquist sampled array that has its elements in bit-reversed order. The technique is generally known as the "fast Fourier transform."

FWAL: This routine calculates Walsh transforms of a Nyquist sampled array. The technique is generally known as the "fast Walsh transform."

AUTCOR: This routine calculates the autocorrelation function for a Nyquist sampled array. From the autocorrelation function, the power spectrum is calculated by the method of refined spectral densities.

PLOTS: This routine plots an equispaced array.

SAVE: Utility subroutine for saving arrays.

UNSAVE: Utility subroutine for retrieving saved arrays.

47

3.2 Input Data Descriptions

Listed below are the input cards and their format.

CARD 1: READ MTRACE

FORMAT (I5)

MTRACE: The total number of the sets of data to be processed by the program.

CARD 2: READ FMT

FORMAT (20A4)

FMT: An array that contains the format of the input data arrays $(X(I), Y(I))$.

CARD 3: READ TITLE

FORMAT (20A4)

TITLE: An array that contains 80 characters of alpha-numeric information describing the data.

CARD 4: READ NROT, NSTAR, NPOW, ND, JD.

FORMAT (5I5)

NROT: If 0, then the input array must be rotated.

NSTAR: The number of points in the equispaced Nyquist array. It must be an interger power of two.

NPOW: NSTAR = 2^{**} NPOW.

ND: Used for plotting power spectra, every ND^{th} point will be plotted.

JD: Used for plotting time series, every JD^{th} point will be plotted.

48

CARD 5:   READ INTERP, IFFFT, IFFWAL, IFAUTO, IFGOF

   FORMAT (6I5)

   INTERP = 0 Main calls NYQST

     = 1 Main calls LNYQ

   IFFFT = 0 Main skips FFT

     = 1 Main calls FFT

   IFFWAL = 0 Main skips FWAL

     = 1 Main calls FWAL

   IFAUTO = 0 Main skips AUTCOR

     = 1 Main calls AUTCOR

   IFGOF = 0 Main calls FILTER

     = 1 Main calls GNFILT.

CARD 6:   READ YSCAL, XSCAL

   FORMAT (2(2X, E13.7))

   YSCAL: A scale factor that multiplies the input amplitude
     array.

   XSCAL: A scale factor that multiplies the input time
     values.

   Note: The program assumes the time array is in nanoseconds
     after it is scaled.

CARD 7:   READ JF

   FORMAT (2X, I4)

   JF: The number of points in the input time and amplitude
     arrays.

CARD 8:   READ (XF(I), YF(I), I=1, JF)

            FORMAT (FMT)

            XF: An array which contains the initial time array.

            YF: An array which contains the initial amplitude array.

CARD 9:   READ XZ YZ FORMAT (2(1X, E11.4))

            XZ,YZ: Are the tablet coordinates of the zero point of the
                     trace.

CARD 10:   READ X1, X2, Y1, Y2

            FORMAT (4F510)

            X1, X2: Coordinate labels for digitization points on the X
                     axis of the grid.

            Y1, Y2: Coordinate labels for digitization points on the
                     Y axis of the grid.

CARD 11:   READ (XT(I), YT(I), I=1,4)

            FORMAT (2(1X, E11.4))

            XT,YT: Tablet coordinates for the four grid measurements
                     defined by X1, X2, Y1, and Y2.

CARD 12:   READ T, V

            FORMAT (2(1X, E11.4))

            T: A time-scale factor nanoseconds/division.

            V: A voltage scale factor volts/division.

            Note: Cards 9, 10, 11, and 12 only appear if NROT = 1.

CARD 13:   READ F1, F2, GW2

            FORMAT (3E13.7)

            F1: Half-power point for the low-pass digital filter given
               in megahertz.

F2: <Fl is another frequency at which we specify the gain, GW2, and thus determine the number of poles for the filter. F2 is in megahertz and 0 < 0.5.

Note: Card 13 only appears if IFGOF=1. If IFGOF=0 then

CARD 13: READ   NP

FORMAT(IS)

CARD 14: READ(A(D), K=1, NP)

FORMAT (5(2X, El3.7))

READ (B(K), K=1, NP+1)

FORMAT (5(2X,El3.7))

If further analysis is desired repeat cards 2 through 12. The program will continue analyzing data until all input data are exhausted. If much similar data are to be processed, it would be convenient to define all the parameters internal to the program and then delete a great many input cards.

A computer printout of the signal analysis program is included as appendix A.

4. REPRESENTATION OF EMP WAVEFORMS BY PARAMETRIZED FUNCTIONS

As previously reported, it is very convenient to characterize an EMP-induced waveform by a finite set of parametrized functions. This allows an analyst to conveniently handle the thousands of waveforms generated in any given experiment. At present, most analysts deal with the data as a digital record consisting of $\sim 10^3$ digital values per waveform and an equivalent number of digital values in the Fourier power spectrum. Although this is a valid approach, it is difficult and tedious to make comparisons between large sets of data and to discover trends in the data.

A large fraction of the data that is generated in EMP experiments consists of waveforms that can be described as a product of a growth factor, an exponential damping factor, and a sinusoidal function. In many of these records there is more than one dominant frequency. Most data are then fitted to a judicious mix of the functions ϕ_n, where

$$\phi_n = t^{n-1} \exp(-\xi_n t)\sin\omega_n t \, . \tag{76}$$

51

A linear combination of the Φ_n generally suffices, thus a function Φ is fit to the digital data, where

$$\Phi = \sum_{n=0}^{N} a_n \Phi_n \quad ; \tag{77}$$

or

$$\Phi = \sum_{n=o}^{N} a_n \Phi_n + \sum_{n'=o}^{N'} b_{n'} \Phi_{n'} \tag{78}$$

There is no general rule that allows a blind selection of such functions; thus, at some point the data must be examined and a "guess" made for a good set of functions to characterize the data.

Following is a description of the method used to fit an N parameter nonlinear function to a set of data. Also, some examples of data are included, which were processed by using the computer codes especially developed for this problem. A listing of the computer program with detailed annotations is given in appendix B.

4.1 Method of Nonlinear Least Squares

For simplicity, consider the problem of fitting an N-parameter function with one independent variable X_i, $\Phi(X_i, P_1 \ldots P_N)$, to the measured quantities $Y(X_i)$ where the subscript i refers to the ith data point. Thus, we find the parameters that minimize

$$S^2 = \sum_{i=1}^{M} \left[Y\left(X_i\right) - \Phi\left(X_i, P_1 \ldots, P_n\right) \right]^2 W_i \quad , \tag{79}$$

where M is the number of data points, and W_i is the statistical weight of the ith data point.

Let us assume now that an iterative procedure has been defined for determining the parameter P_j; that is, P_j^k = the kth iteration of the jth parameter. Expand the function Φ in a Taylor series about P_j and truncate all but the linear terms. Then use the kth iteration to determine the parameters for the $(k+1)^{th}$ iteration. Let

$$\Phi = \Phi^k + \sum_{j=1}^{N} \left[\frac{\partial \Phi}{\partial P_j}\right]^k \Delta P_j^k \quad , \tag{80}$$

where

$$\Delta P_j^k = P_j^{k+1} - P_j^k \quad ,$$

and

$$\Phi^k = \Phi\left(X_i, P_1^k, \ldots, P_N^k\right) \, .$$

Note that given P_j^k, we must now determine P_j^{k+1}. Substituting equation (80) into equation (79) yields in the linear approximation

$$S^2 \approx S_k^2 = \sum_{i=1}^{M} \left(Y\left(X_i\right) - \Phi^k - \sum_{j=1}^{N} \left[\frac{\partial \Phi}{\partial P_j}\right]^k \Delta P_j^k\right)^2 W_i \, . \tag{81}$$

To minimize S_k^2, form

$$\frac{\partial S_k^2}{\partial P_\ell} = 0, \quad \ell = 1, \ldots, N, \tag{82}$$

which yields

$$\sum_{i=1}^{M} \left[Y\left(X_i\right) - \Phi^k\right]\left[\frac{\partial \Phi}{\partial P_\ell}\right]^k W_i = \sum_{i=1}^{M} \left[\sum_{j=1}^{N} \left[\frac{\partial \Phi}{\partial P_j}\right]^k \Delta P_j^k \left[\frac{\partial \Phi}{\partial P_\ell}^k\right]\right]_i , \quad \ell = 1, \ldots, N. \tag{83}$$

53

Defining $\psi_i^k = Y\left(X_i\right), - \phi^k,$

$$z_{\ell i}^k = \left[\frac{\partial \Phi}{\partial P_\ell}\right]^k ,$$

$$C_\ell^k = \sum_{i=1}^M \psi_i^k \, z_{\ell i}^k \, W_i \; ; \tag{84}$$

then from equations (81) and (82),

$$
\begin{bmatrix}
\displaystyle\sum_{i=1}^M \left(z_{1i}^k \, z_{1i}^k\right)W_i, \ldots, \sum\left(z_{1i}^k z_{Ni}^k\right)W_i \\
\cdot \qquad\qquad \cdot \\
\circ \quad \circ \qquad \quad \circ \\
\circ \qquad \circ \qquad \quad \circ \\
\qquad \cdot \\
\displaystyle\sum_{i=1}^M \left(z_{N_i}^k z_{1i}^k\right)W_i, \ldots, \sum\left(z_{N_i}^k z_{N_i}^k\right)W_i
\end{bmatrix}
\begin{bmatrix}
\Delta P_1^k \\
\Delta P_2^k \\
\circ \\
\circ \\
\cdot \\
\Delta P_N^k
\end{bmatrix}
=
\begin{bmatrix}
C_1^k \\
C_2^k \\
\circ \\
\circ \\
\cdot \\
C_N^k
\end{bmatrix}
\tag{85}
$$

Or, more concisely,

$$\underset{\sim}{A} \cdot \Delta\vec{P} = \vec{C} \quad . \tag{86}$$

Equation (83) can now be inverted to solve for the ΔP_j^k and, hence, the P_j^{k+1}. If now our iterative procedure is converging, the P_j^{k+1} should be closer to the values P_j^* which minimizes S^2 and can then be used for the next iteration. This iterative procedure is continued until the use of P_j^{k+1} produces a chi-squared which differs from that using P_j^k by less than some preset value, V usually $V = 10^{-6}$; that is,

$$\left|S_{k+1}^2 - S_k^2\right| < V \quad . \tag{87}$$

54

To insure that the step ΔP_j^k is converging, first note that in the linear approximation the quantity

$$D_T^k = \sum_{\ell=1}^{N} C_\ell^k \, \Delta P_\ell^k = \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{\ell=1}^{N} z_{ji}^k \, \Delta P_j^k z_{\ell i}^k W_i \Delta P_\ell^k , \tag{88}$$

$$= \sum_{i=1}^{M} W_i \left[\sum_{j=1}^{N} z_{ji}^k \, \Delta P_j^k \right]^2 \geq 0 ,$$

which implies the D_T^k must always be positive semidefinite. Thus, if at the kth iteration $D_T^k < 0$, then change the sign of all the ΔP_ℓ^k. Another approach that shows this is to expand S^2 in a Taylor series and to keep only the first deriviatives of ϕ; that is,

$$S^2 = S_k^2 \bigg|^k + \sum_\ell \frac{\partial S^2}{\partial P_\ell} \bigg|^k \Delta P_\ell^k + \frac{1}{2} \sum_{\ell j} \frac{\partial S^2}{\partial P_\ell \partial P_j} \bigg|^k \Delta P_\ell^k \Delta P_j^k + \ldots ,$$

$$= \sum_{i=1}^{M} W_i \left[\left(\psi_i^k \right)^2 - 2\psi_i^k \sum_\ell z_{\ell i}^k \, \Delta P_\ell^k + \sum_{j\ell} z_{ji}^k z_{\ell i}^k \Delta P_j^k \, \Delta P_\ell^k \right] . \tag{89}$$

If ΔP^k is considered as defining a vector, construct S^2 to be a function of one variable α, for example, by letting $\Delta P_\ell^k \rightarrow \alpha \Delta P_\ell^k$; then we have

$$S^2(\alpha) = S_k^2 - 2\alpha D_T^k + \alpha^2 D_T^k . \tag{90}$$

Now the slope of $S^2(\alpha)$ evaluated at $\alpha=0$ is

$$\frac{dS^2(\alpha)}{d\alpha} \bigg|_{\alpha=0} \simeq - 2D_T^k ; \tag{91}$$

and to reach a minimum this slope must be negative, that is, $D_T^k > 0$, as in equation (86).

Although the condition $D_T^k > 0$ insures convergence, barring roundoff errors, the procedure can take divergent steps, thus oscillating and giving slow convergence. The following test has been used to overcome this problem and found to work satisfactorily on most data. It is first necessary to check D_T^k and, if it is negative, to change the sign of all the ΔP_ℓ^k. Then test to determine if the value $S_{k+1}^2 (\alpha=1) < S_k^2$; if it is, perform the test described by equation (85). If $S_{k+1}^2 [(\alpha) = 1] > S_k^2$, $S_{k+1}^2 (\alpha)$ must be calculated for the following four values of α. Thus,

$$\alpha = 1/2 \ ,$$
$$\alpha = D_T^k / \left(S_{k+1}^2 (\alpha=1) - S_K^2 + 2D_T^k \right)$$

[determined by the parabola with slope $-D_T^k$ at $\alpha=0$ and passing through the points S_k^2 and $S_{k+1}^2 (\alpha=1)$],

$$\alpha = 1/D_T^k$$

(determined from the parabola given in equation (90)),

$$\alpha = \left[S_{k+1}^2 (\alpha=1) + 3S_k^2 - 4S_{k+1}^2 (\alpha=1/2) \right] / 4 \left[S_k^2 + S_{k+1}^2 (\alpha=1) - 2S_{k+1}^2 (\alpha=1/2) \right]$$

[determined by the parabola passing through the points S_k^2, $S_{k+1}^2 (\alpha=1/2)$, and $S_{k+1}^2 (\alpha=1)$] with the restriction that any value of $\alpha < 10^{-2}$ is ignored. From these four (or less) values of $S_{k+1} (\alpha)$, find the minimum, compare it with S^2, and perform the termination test equation (85) if S^2 is improved. If, however, S^2 is still not improved, then either bad data points remain or the initial starting parameters are too far from the ones that minimize S^2.

After achieving a minimum as determined by equation (87), an error analysis is performed to give an indication of how well the parameters can be determined from the data. The method used is that described by Cohen, Crowe, and Dumond.[3] Their discussion deals only with linear least squares fitting and we have not made a detailed study for the nonlinear case. However, in the neighborhood of the minimum in S^2 the linearization given by equation (81) should be a good approximation. With the above warning the error (σ_ℓ) on the ℓth parameter is given by,

$$\sigma_\ell = \sqrt{\left(A^{-1}\right)_{\ell\ell} \chi^2} \ . \tag{92}$$

where

$$\chi^2 = S^2/(M-N), \tag{93}$$

which is the chi-squared normalized with the number of degrees of freedom. To be conservative in our error estimate χ^2 is set equal to one if the fit gives a smaller value.

The other important quantities in the error analysis are the correlation coefficients defined by

$$\rho_{ij} = (A^{-1})_{ij} \Big/ \sqrt{(A^{-1})_{ii} (A^{-1})_{jj}} \tag{94}$$

The correlation coefficients are necessary for computing effects of error propagation when using the "best" parameters. Consider a function f which depends on L fitted parameters, then the error on f is given by

$$\sigma_f = \sum_{\ell=1}^{L} \left(\frac{\partial f}{\partial P_\ell}\right)^2 \sigma_\ell^2 + 2\sum_{i<j} \rho_{ij} \frac{\partial f}{\partial P_i} \ \frac{\partial f}{\partial P_j} \ \sigma_i \sigma_j \tag{95}$$

[3]E. R. Cohen, K. M. Crowe, and J. W. M. Dumond, *Fundamental Constants of Physics*, Interscience Publishers, Inc., New York (1957), Ch. 7.

57

4.2 Parametrization of an EMP Waveform and its Autocorrelation Function

As simple examples of the use of the program the data of figure 4 has been analyzed in several different ways. The power spectrum of the time series data is shown in figure 5. It is seen that there are two predominant frequencies in the time waveform, 13.2 and 24 MHz. To isolate one of the frequencies (and obtain a less complicated time series), the data of figure 4 were digitally filtered, using a 6-pole low-pass digital filter with a half-power point at 17.5 MHz and a gain at 25 MHz of 0.05. The resulting time series is shown in figure 25. Its power spectrum is shown in figure 26. It is seen from figures 25 and 26 that the time series consists primarily of a sinusoid with frequency of 13.2 MHz. To the time series of figure 25 is fit the following function

$$\phi(t,\vec{P}) = P(1) \ t \ \exp \ (-P(2)t)\sin[P(3)t + P(4)] \tag{96}$$

It took the program five iterations to converge to a solution. The initial estimates and the final fitted parameters are given in table II. A combined plot of the experimental data and the fitted function is shown in figure 27. Generally, the fit is good only in the central portion of the trace and it fails badly at the beginning and end of the trace. This clearly means that equation (96) is not the best representation of this trace.

From the filtered time series of figure 25, the autocorrelation (lagged products) function was formed as displayed in figure 28. The following function was fit to these data.

$$\phi(t,\vec{P}) = P(1)\exp[-P(2)t]\cos[P(3)t] \tag{97}$$

It took the program two iterations to converge to a solution. The initial estimates and the final fitted parameters are given in table III. A combined plot of the data and the fitted functions is shown in figure 29. As can be seen the fit is quite good over the entire trace.

As our third example, the autocorrelation function for the data of figure 4 was calculated and is plotted in figure 30 (no filtering of any kind was done); it was fit by the following function,

$$\phi(t,\vec{P}) = P(1)\exp[-P(2)t]\cos[P(3)t] + P(4)\exp[-P(5)t]\cos[P(6)t] \tag{98}$$

58
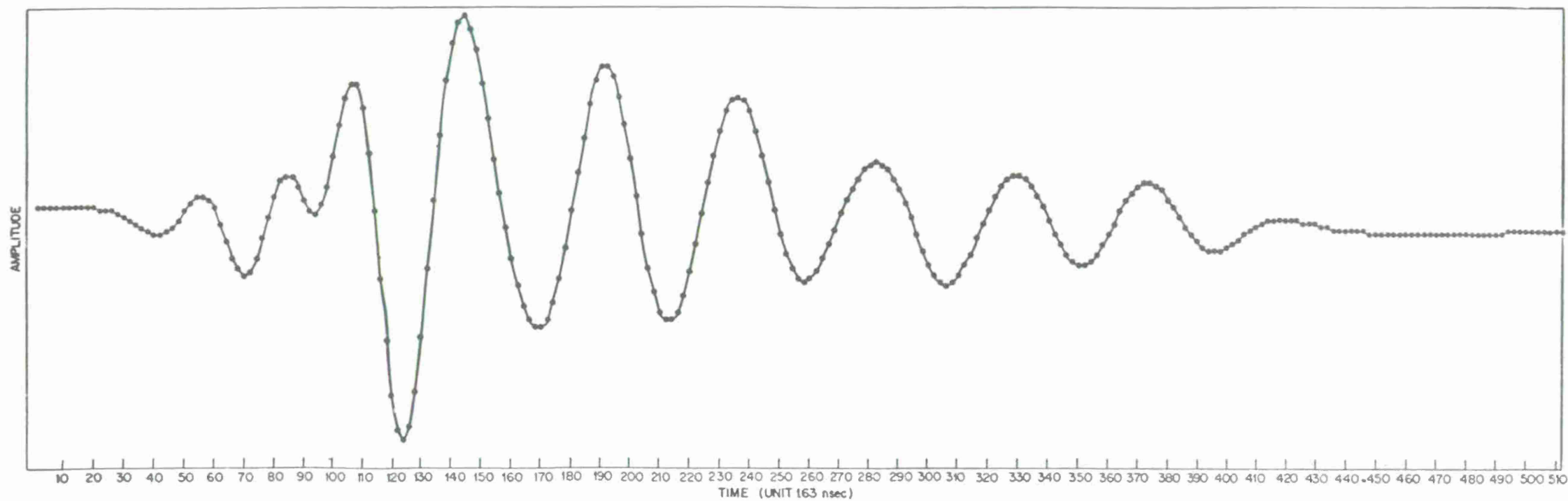
Figure 25. Time series of figure 4 after digital filtering.

59
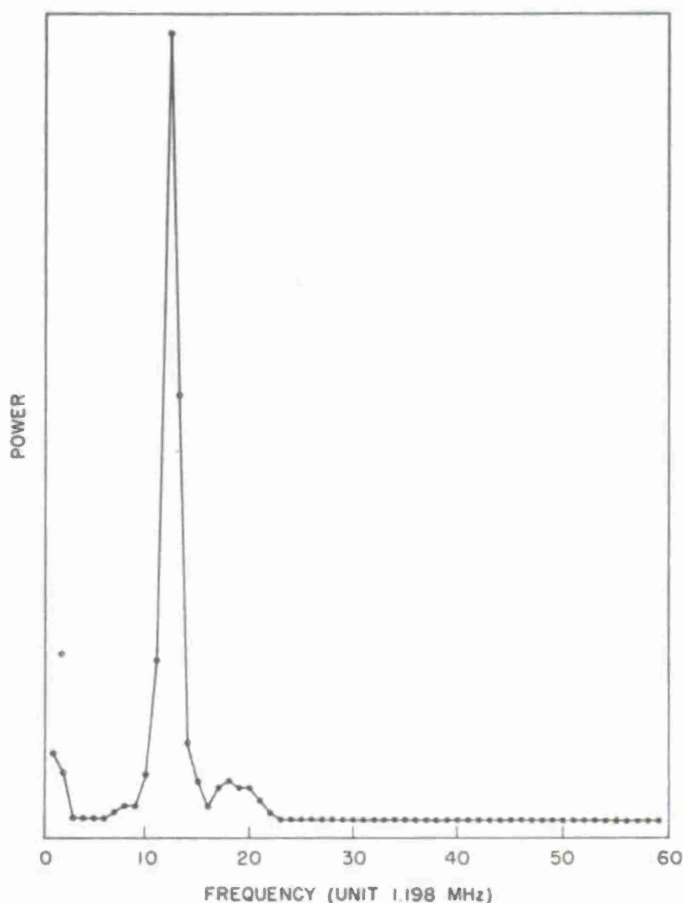
Figure 26.   Power spectrum for the time series of figure 6.

It took the program three iterations to converge to a solution.
The initial estimate and the final fitted parameters are given in
table IV. A combined plot of the experimental data and the fitted
function is shown in figure 31. The fit is quite excellent throughout
the length of the trace.

Several attempts were made to fit the data of figure 4 with
various forms of the functions listed in equation (76). Whenever n was
greater than 2, the fitting procedure either diverged or produced a bad
fit.

TABLE II.   INITIAL ESTIMATES AND FINAL FITTED PARAMETERS FOR EQUATION (96)
            AND DATA OF FIGURE 25

Parameter	Initial estimates	Final fitted values
P(1)	1.08	+0.742
P(2)	4.27	+4.958
P(3)	90.43	+84.433
P(4)	-19.59	-18.207

TABLE III.   INITIAL ESTIMATES AND FINAL FITTED PARAMETERS FOR EQUATION (97)
             AND DATA OF FIGURE 28

Parameter	Initial estimates	Final fitted values
P(1)	+0.001	+0.00102
P(2)	+4.27	+3.921
P(3)	+86.35	+84.196

TABLE IV.   INITIAL ESTIMATES AND FINAL FITTED PARAMETERS FOR EQUATION (98)
            AND DATA OF FIGURE 30

Parameter	Initial estimates	Final fitted values
P(1)	0.0014	0.00108
P(2)	3.92	4.0289
P(3)	84.20	84.7402
P(4)	0.001	0.00148
P(5)	3.92	9.8083
P(6)	147.58	146.0411
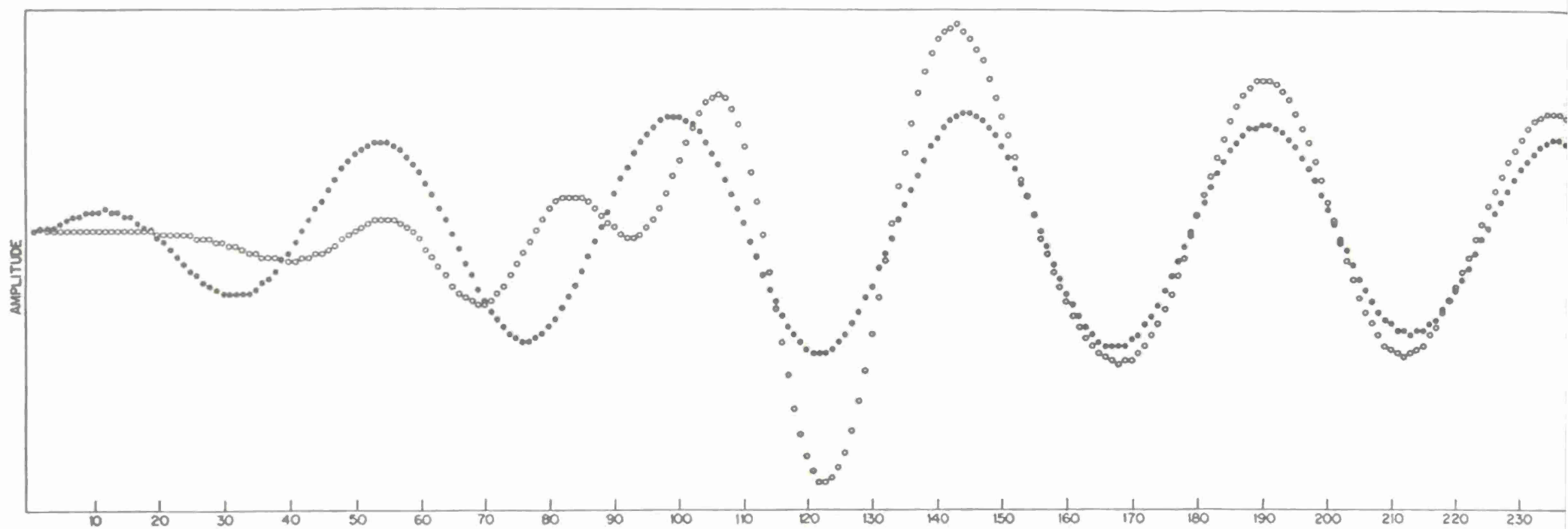
Figure 27.   Plot of a least-squares fit of equation (96) to the time
series of figure 6.

240 250 260 270 280 290 300 310 320 330 340 350 360 370 380 390 400 410 420 430 440 450 460 470 480 490 500 510

TIME (UNIT 1.63 nsec)

TIME (UNIT 1.63 nsec)
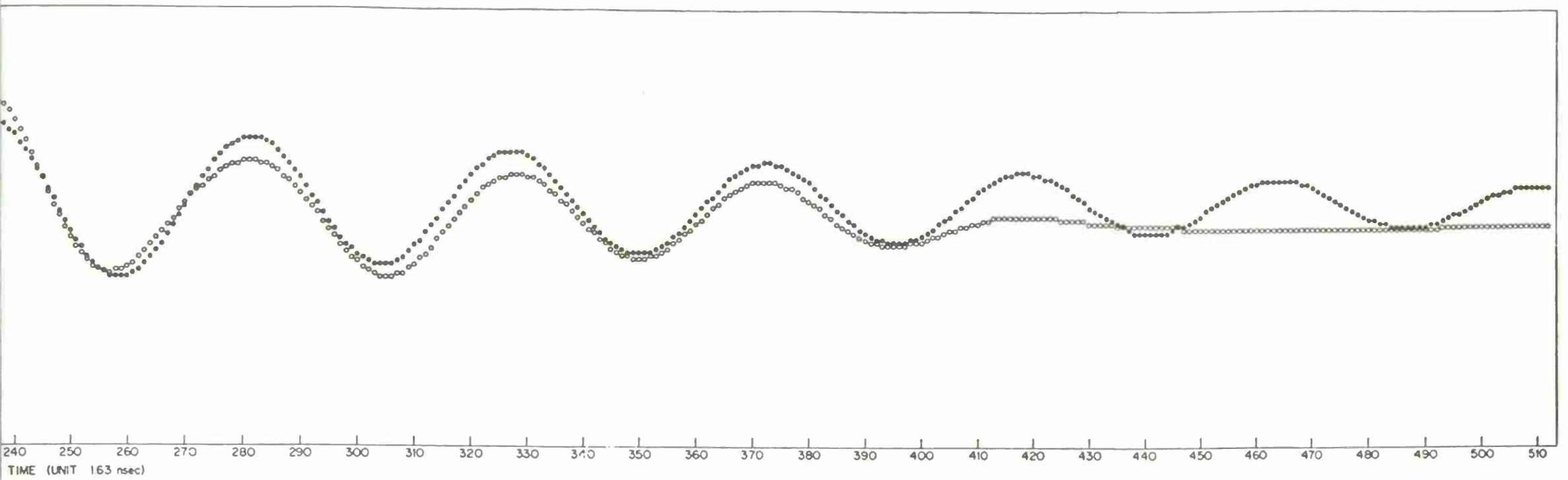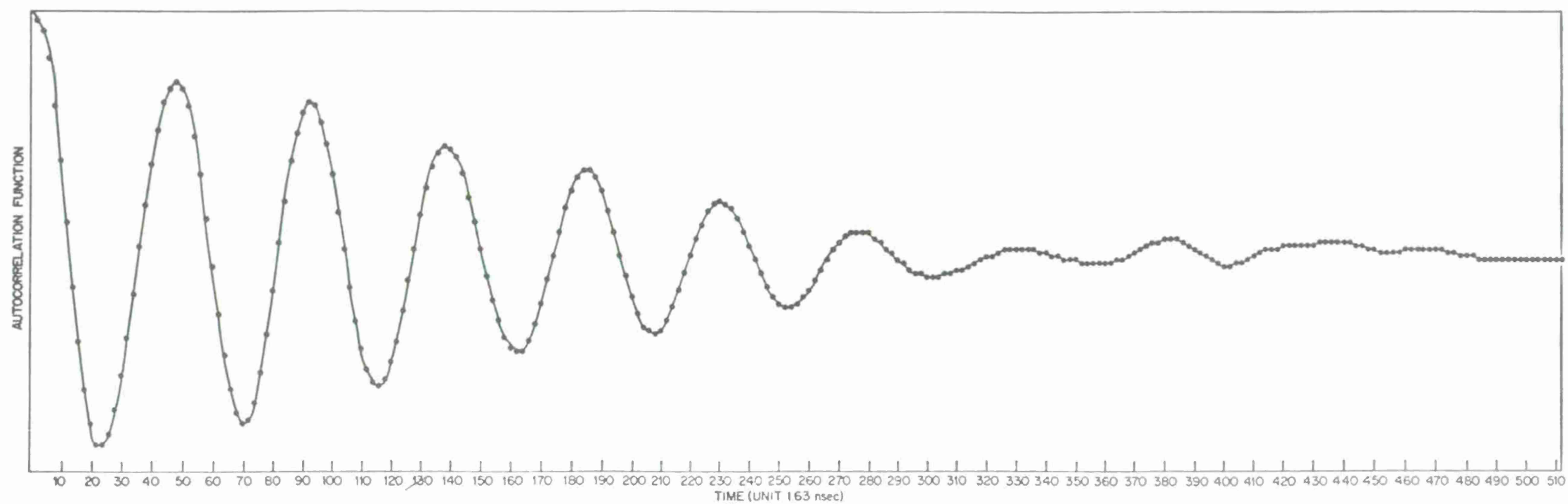
Figure 28. Autocorrelation function for the time series of figure 6.
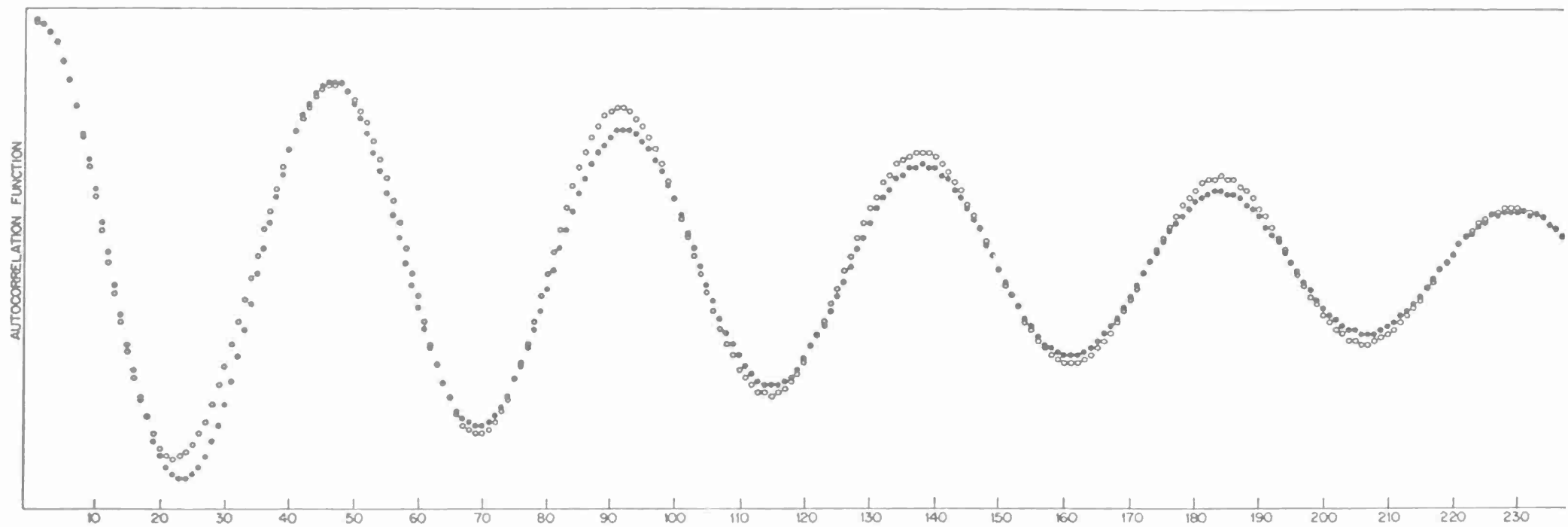
Figure 29. Plot of a least-squares fit of equation (97) to the auto-
correlation function of figure 21.

TIME (UNIT 1.63 nsec)
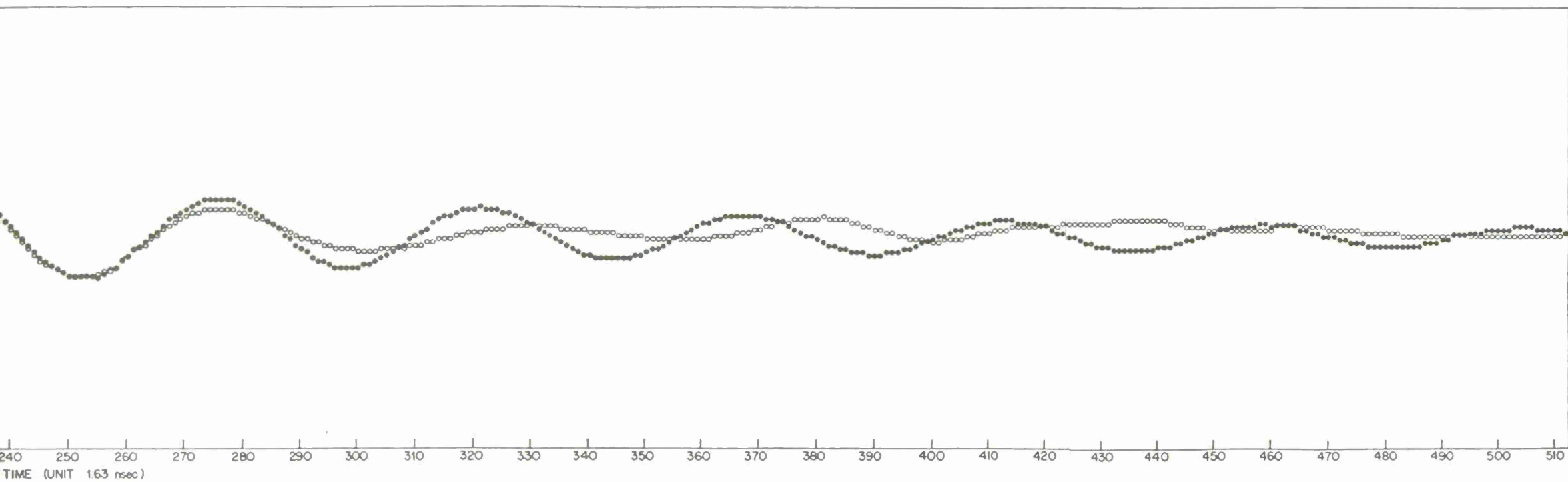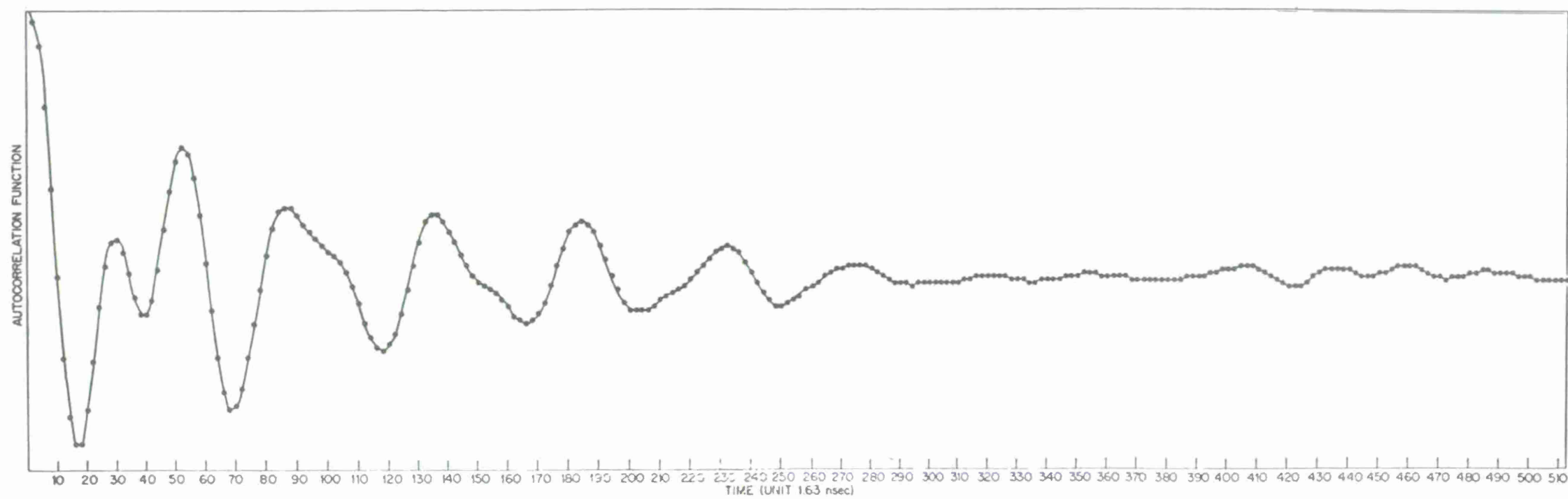
Figure 30. Autocorrelation function for the time series of figure 1.
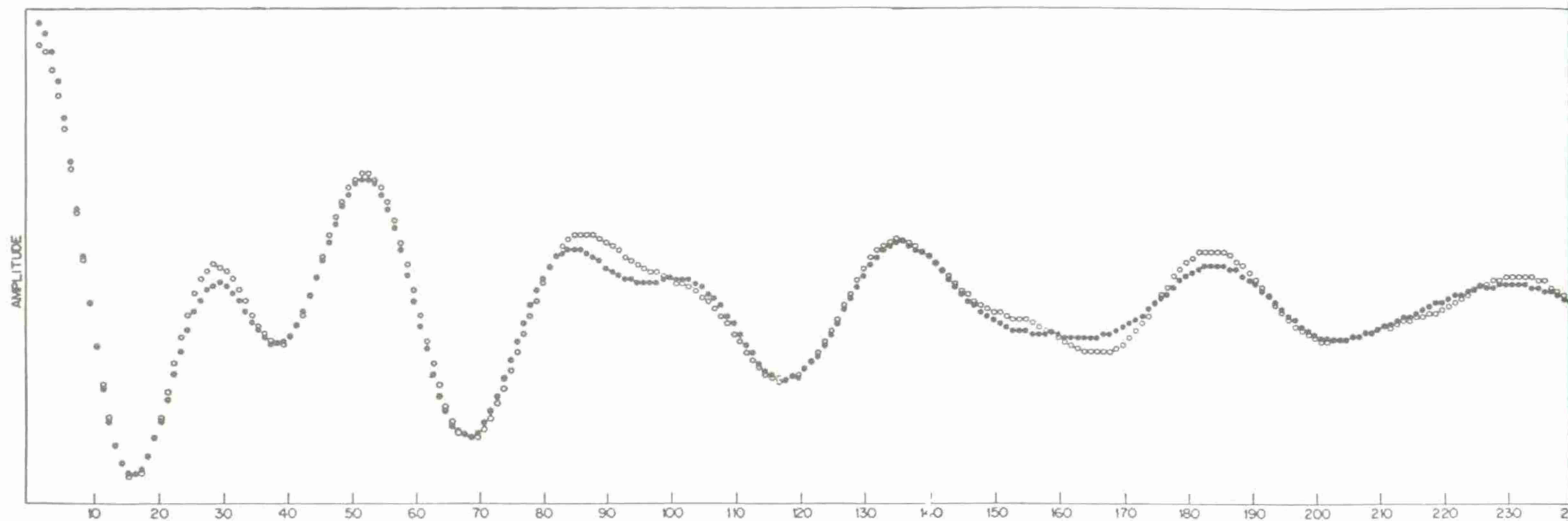
Figure 31. Plot of least-squares fit of equation (98) to the auto-
correlation function of figure 11.

71a

TIME (UNIT 1.63 nsec)
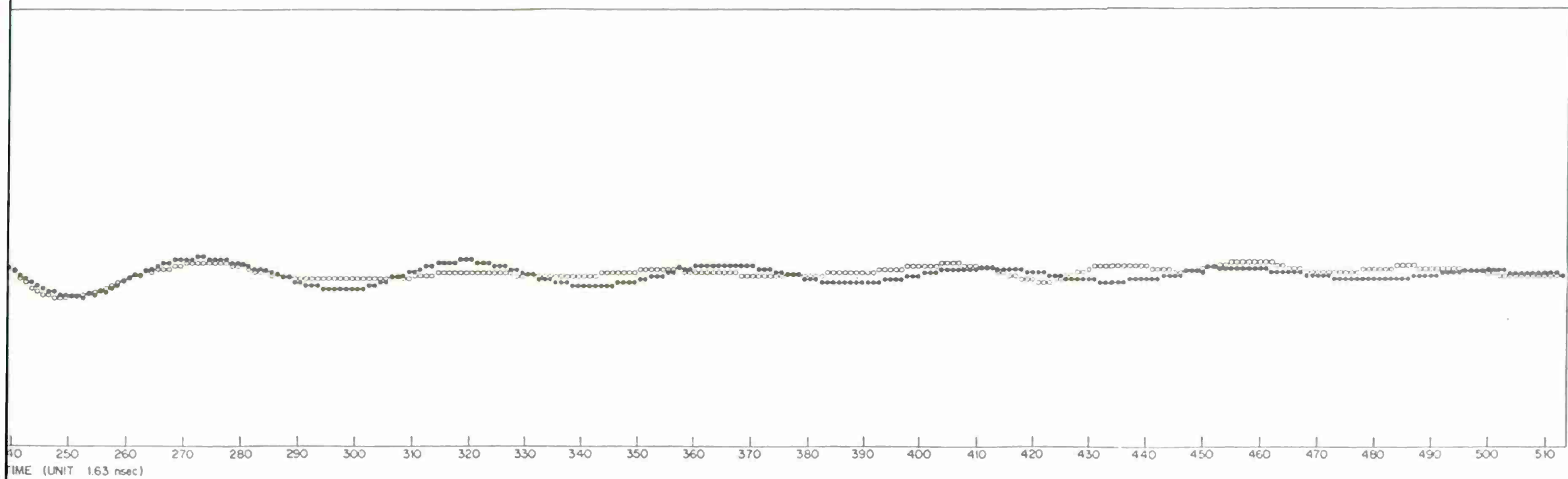
It seems clear that before fitting a function to the data (to obtain parametrization) a process such as digital filtering or formation of the autocorrelation function is necessary to isolate the functional dependence (at least the oscillatory part).

Following are parameters to the power spectrum of the record. Since the power spectrum, $P(\omega)$, is given by the cosine transform of the autocorrelation function, we have

$$P(\omega) = 2 \int_0^\infty \cos\omega t \, [P(1)\exp\{-P(2)t\}\cos[P(3)t]$$
$$+ P(4)\exp[-P(5)t]\cos\{P(6)t\}]dt \quad . \tag{99}$$

Thus, after some algebraic manipulation, a power spectrum is given by

$$P(\omega) = \frac{P(1)\ P(2)}{[\omega-P(3)]^2+P(2)^2} + \frac{P(4)\ P(5)}{[\omega-P(6)]^2+P(5)^2} \quad . \tag{100}$$

It would have been just as easy to fit equation (100) to the power spectrum displayed in figure 5 and thus obtain a parametrization of the data. However, we have used a plot of equation (100) superimposed on the power spectrum of figure 5 as shown in figure 32.

Considering the storage of data, it is concluded that only six numbers must be stored to characterize the pulse instead of thousands. Thus, long-term storage costs are minimized and a high degree of analytical ease is realized in handling large blocks of data.

4.3 Program Listing and Description

The program is written in FORTRAN and consists of eleven subroutines and two user supplied functions; the program names with a description of each are listed below.

MAIN   This routine directly or indirectly calls the rest of the subroutines and thus controls the passage of the program through all the calculations used in the fit.

READIN This routine reads in all the input parameters and data as well as setting up all the internal control parameters.

LSQPHI This routine calculates the theoretical function, residuals, and chi square. The user must supply the function PHIFNC(I) for LSQPHI to use.

73
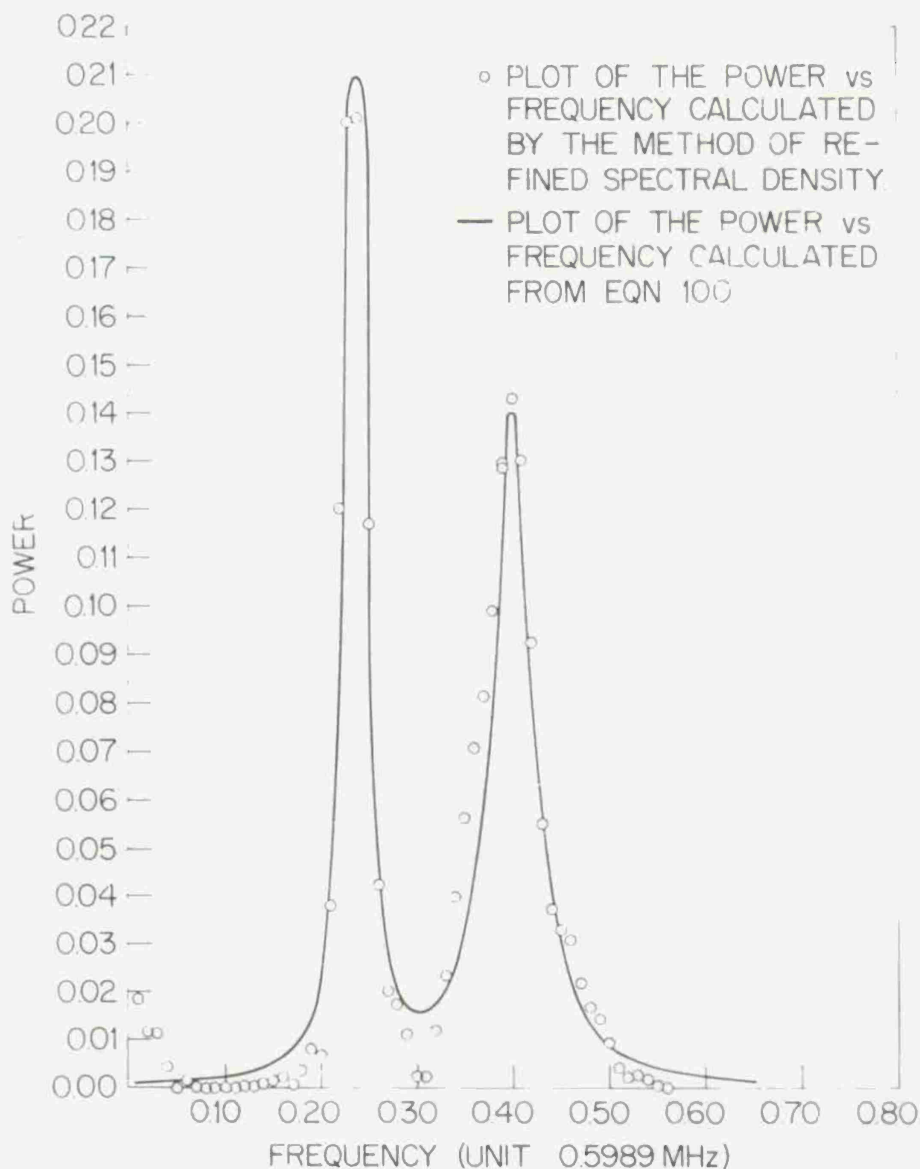
Figure 32. Plot of power spectrum obtained from equation (100) and that
obtained from a numerical algorithm.

ELTS(I)   This routine calculates all the derivatives of the
          theoretical function at the Ith data point. The user must
          supply the function (DPHI(I)J) for ELTS to call.

GRIND   This routine sets up the matrix of derivatives, the
          constants in the normal equations, then calls the matrix
          inverter.

74

TESTS   This routine does the necessary testing of the changes in the parameters and performs the necessary changes.

ERRMAT   This routine does all the calculations for a full error analysis on the fitted parameters.

CALS2   This routine is called by TESTS repeatedly for the purpose of swapping arrays and computing chi.

INVMAT   Inverts a symmetric N x N matrix.

SCRIBE   This routine writes out all the results of the program fit.

PLOTS   Gives a plot of the experimental data and theoretical function on the same graph so that the goodness of the fit can be clearly seen

PHIFNC(I) $PHIFNC = \phi(X(I), P(1), P(2),...,P(N))$

DPHI(I,J) $DPHI = \dfrac{d\phi}{dP(J)} (X(I), P(1), P(2),...,P(N))$

The flow chart of figure 33 shows the logical connection of the subroutines and functions.

Listed below are the input parameters, their meaning, and their format.

CARD 1   READ   JSTOP,FMF FORMAT (I5,5A4)

        JSTOP   Maximum number of iterations allowed for each fit. This is used to prevent excessive iterations in case there are errors in the experimental data or control parameters. Usually JSTOP $\sim$ 100.

        FMT   The format with which the experimental data are to be read in, e.g., (6E13.7).

        NOTE: Card 1 appears once independent of the number of sets of data to be fit.

CARD 2   READ   TITLE FORMAT (20A4)

        TITLE   Any combination of alphanumeric data may appear in columns 1 through 80 to identify the fit.
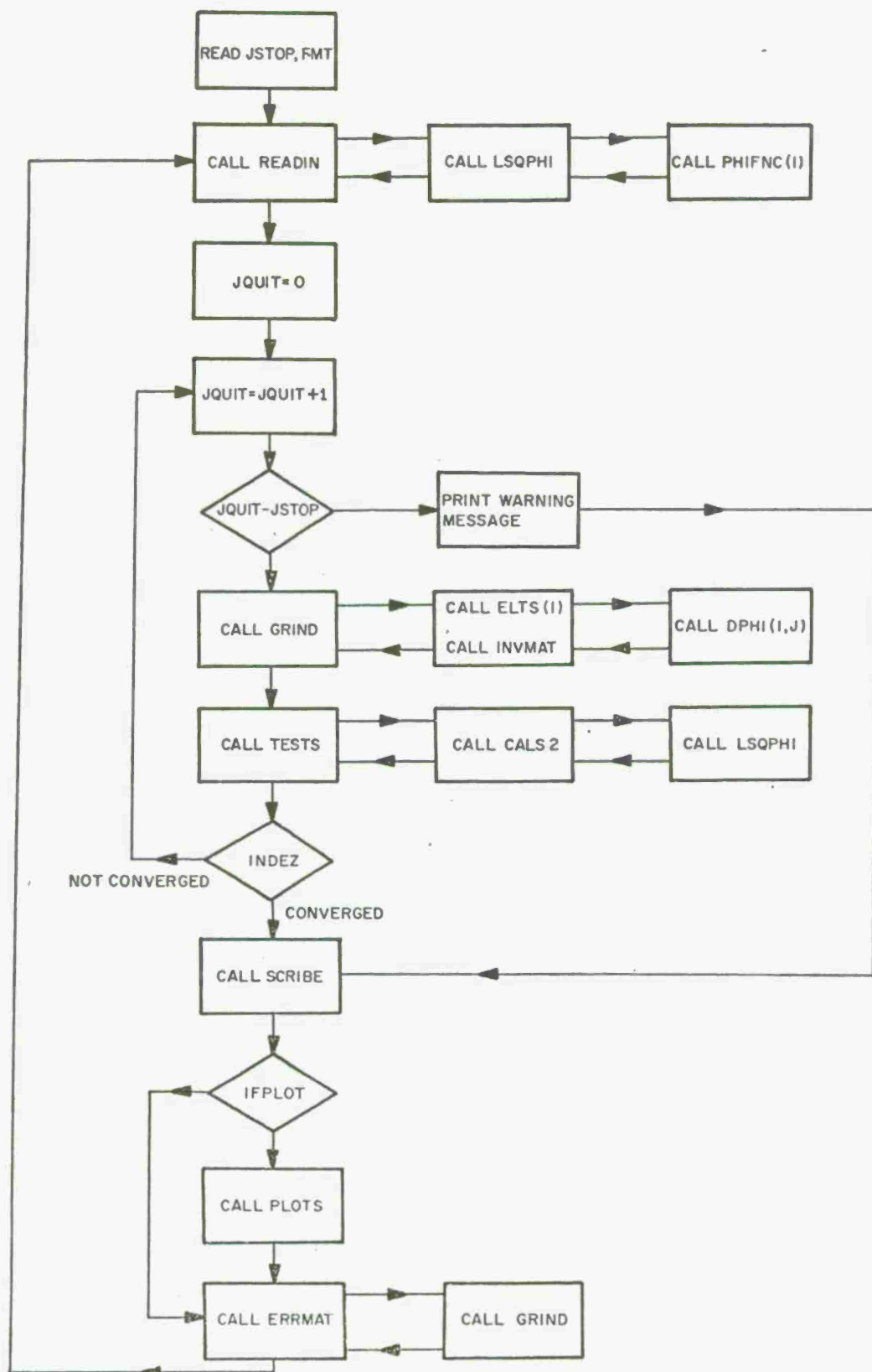
Figure 33. Flow chart of nonlinear least-squares fitting program.

CARD 3   READ   NCHAN, NBADPT, NRUNS, NRP, NPSTEP, IFPLOT, IFCOR
          FORMAT (7I5)

     NCHAN   The number of experimental data points.

     NBADPT   The number of bad points in the data.

     NRUNS   The number of sets of experimental data to be added
              together.

       NRP   The number of independent parameters.

     NPSTEP ≥ 0, used for plotting; that is, every NPSTEPth data
              point is plotted starting with the first (0 is
              internally set to 1).

     IFPLOT   = 0 no plot of data and fit
              = 1 gives plot

      IFCOR   = -1 This prints and punches the correlation
                    coefficients
              = 0   Does not print or punch
              = 1   Only prints

  CARD 4   READ   (P(I), I=1, NP)

       P   the array containing the fitted parameters.

CARDS 5   READ   (Y(I), I=1, NCHAN) FORMAT (FMT)

       Y   the array containing the experimental data, one run
            following another for NRUNS worth of data.

CARDS 6   READ   (X(I), I=1, NCHAN) FORMAT (FMT)

       X   the array containing values of the dependent
            variable for ϕ and Y; for example, $\phi(X(I),\vec{P})$.

CARDS 7 READ   (W(I), I=1, NCHAN) FORMAT (FMT)

       W   the array containing values of the statistical
            weights for the Ith data point.

CARDS 8   READ   (NBAD(I), I=J, NBADPT)

                NBAD   Index number of the bad data points in any order.
                        These cards do not appear in the data deck if
                        NBADPT = 0.

     If multiple fits are desired, repeat cards 2 through 8.   The
program will continue doing fits until it has exhausted all of the input
data.

(JAN 73)   OS/360   FORTRAN H

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                  SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF

```
      C******************************************************************************
      C
      C MAIN PROGRAM FOR THE ANALYSIS OF DIGITAL TIME SERIES
      C PROGRAMMER DR. THOMAS A TUMOLILLO
      C             USAMC HARRY DIAMOND LABORATORIES
      C             WASHINGTON8D.C. 20438
      C
      C COMMON VARIABLES DEFINITION
      C
      C     YVYQA    COMPLEX ARRAY USED TO STORE THE DIGITAL TIME SERIES AT VARIOUS
      C              STAGES OF THE ANALYSIS.
      C
      C   NW         SYMBOLIC DESIGNATOR FOR THE OUTPUT LINE PRINTER
      C   VR         SYMBOLIC DESIGNATOR FOR THE INPUT CARD READER
      C     NT       SYMBOLIC DESIGNATOR FOR THE OUTPUT CARD PUNCH
      C     XF       ARRAY WHICH INITIALLY CONTAINS THE TIME VALUES
      C     YF       ARRAY WHICH INITIALLY CONTAINS THE AMPLITUDE VALUES
      C     NSTAR    THE NUMBER OF INTERPOLATED POINTS9IT MUST BE A POWER OF TWO
      C     NPOW     NSTAR=2**NPOW
      C     ND       USED FOR PLOTTING POWER SPECTRA3EVERY ND-TH POINT WILL BE PLOTTED
      C     JF       THE NUMBER OF POINTS IN THE INPUT DIGITAL TIME SERIES
      C     CT       THE COSINE OF THE ROTATION ANGLE
      C     ST       THE SINE OF THE ROTATION ANGLE
      C     XJ       X-COORDINATE OF THE GRATICULE ORIGIN
      C     YJ       Y-COORDINATE OF THE GRATICULE ORGIN
      C     XS       SCALE FACTOR TABLET INTEGERS/SCALE DIVISION ALONG X AXIS
      C     YS       SCALE FACTOR TABLET INTEGERS/SCALE DIVISION ALONG Y AXIS
      C     T        SCALE FACTOR NANOSECONDS/SCALE DIVISION
      C     V        SCALE FACTOR VOLTS/SCALE DIVISION
      C     XVYQ     THE NYQUIST SAMPLING INTERVAL OR THE TIME INTERVAL BETWEEN
```

79

```
C              BETWEEN DIGITAL VALUES
C     XZ       THE X-COORD OF THE ZERO POINT OF THE TRACE
C     YZ       THE Y-COORD OF THE ZERO POINT OF THE TRACE
C     JD       USED FOR PLOTTING TIME SERIES,EVERY JD-TH POINT WILL BE PLOTTED
C     INTERP   INTERP=0 CALL NYQST,=1 CALL LNYQ
C     IFFFT    IFFFT=0 NO FFT,=1 CALL FFT
C     IFFWAL   IFFWAL=0 NO WALSH TRANSFORM,=1 CALL FWAL
C     IFAUTO   IFAUTO=0 NO FT VIA AUTOCORRELATION,=1 CALL AUTCOR
C     IFGOF    IFGOF=0 CALL FILTER,=1 CALL GNFILT
C     IFPUNF   IFPUNF=1 PUNCH OUT FREQUENCY AND POWER SPECTRA VALUES,=0 DONT
C     IFPUNW   IFPUNW=1 PUNCH OUT SEQUENCY AND WALSH POWER VALUES,=0 DONT
C     FPUNA    IFPUNA=1 PUNCH OUT TIME AND AUTOCORRELATION VALUES,=0 DONT
C     IFPUND   IFPUND=1 PUNCH OUT TIME AND AMPLITUDE VALUES OF INPUT DATA,=0 DON
C     IFPUNI   IFPUNI=1 PUNCH OUT TIME AND INTERPOLATED VALUES,=0 DONT
C     IFPUNR   IFPUNR=1 PUNCH OUT FREQUENCY AND REFINED SPECTRAL DENSITIES,=0DON
      COMMON YNYQA(2048)
      COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
     1 XS,YS,T,V,XNYQ
     2,NT,XZ,YZ,JD
      COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
     1IFPUND,IFPUNI,IFPUNR
      COMPLEX YNYQA
      NW=6
      NR=5
      NT=7
      READ (NR,1)MTRACE
    1 FORMAT(I5)
      DO 120 I=1,MTRACE
      CALL READIN
      CALL CSTOUT
      IF(INTERP) 10,10,20
   10 CALL NYQST
      GO TO 30
   20 CALL LNYQ
```

```
30        CALL PLOTS(1)
          CALL SAVE(0)
          NPOWS=NPOW
          NKEEP=NSTAR
          IF(IFGOF) 40,40,50
40        CALL FILTER
          CALL SAVE(1)
          GO TO 60
50        CALL GNFILT
          CALL SAVE(1)
60        CALL PLOTS(4)
          IF(IFFFT) 80,80,70
70        CALL SRTFUR
          CALL FFT
          CALL PLOTS(2)
80        IF(IFAUTO) 100,100,90
90        CALL AUTCOR(1,NKEEP,NPOWS)
100       IF(IFFWAL) 120,120,110
110       CALL UNSAVE(1)
          CALL FWAL
          CALL PLOTS(2)
120       CONTINUE
          STOP
          END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                 SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE READIN
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,X0,Y0,
         1  XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          DIMENSION FMT(20),TITLE(20)
          READ(NR,1)FMT
    1     FORMAT(20A4)
          READ(NR,1) TITLE
          WRITE(NW,2) TITLE
    2     FORMAT(1H1,20X,20A4)
          READ(NR,3) NROT,NSTAR,NPOW,ND,JD
    3     FORMAT(5I5)
          WRITE (NW,13)
   13     FORMAT (1X,'NROT NSTAR NPOW    ND    JD')
          WRITE(NW,14) NROT,NSTAR,NPOW,ND,JD
   14     FORMAT(I4,4I5,/)
          READ(NR,3) INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF
          WRITE(NW,11)
   11     FORMAT(1X,'INTERP IFFFT IFFWAL IFAUTO IFGOF')
          WRITE(NW,12)INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF
   12     FORMAT(1X,2I6,2I7,I6,6I7,/)
          READ(NR,8) YSCAL,XSCAL
    8     FORMAT(2(2X,E13.7))
          WRITE (NW,15)
   15     FORMAT(1X,' Y SCALE FACTOR X SCALE FACTOR ')
          WRITE (NW,16) YSCAL,XSCAL
   16     FORMAT(2(2X,E13.7),/)
          WRITE(NW,5)
```

```
5        FORMAT(' IMAGE OF INPUT DATA')
         READ(NR,7) JF
7        FORMAT(2X,I4)
         READ(NR,FMT) (XF(I),YF(I),I=1,JF)
         DO 100 I=1,JF
         YF(I)=YSCAL*YF(I)
100      XF(I)=XSCAL*XF(I)
         XVYQ=XF(JF)/NSTAR
         WRITE(NW,4) (XF(I),YF(I),I=1,JF)
4        FORMAT(9(1X,E11.4))
         IF(NROT-1)60,50,50
50       CALL SCARTP
         CALL ROT
         WRITE(NW,6)
6        FORMAT(/,' SCALED AND ROTATED DATA')
         WRITE(NW,4) (XF(I),YF(I),I=1,JF)
60       RETURN
         END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                    SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE GNFILT
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1  XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA,U,R,S,ROOT,BETA,UNIT
          EQUIVALENCE (ARRAY(1),YNYQA(1))
          DIMENSION ARRAY(4096)
          DIMENSION STH(40),CTH(40),INOM(40)
          DIMENSION U(40),R(20),S(20),ROOT(20),P(20),
         1Q(20),PQ(20),A(20),B(20)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC(
C
C THIS ROUTINE WILL CALCULATE THE COEFFICIENTS A(J) AND B(J)
C FOR A BUTTERWORTH LINEAR RECURSIVE FILTER. IF Y(K) AND U(K)
C ARE THE OUTPUT AND INPUT ARRAYS FOR THE FILTER THEN,
C
C    Y(K)=-SUM(1 TO N)(A(J)*Y(K-J))+SUM(1 TO N+1)(B(J)*U(K-J+1))
C
C THE GAIN FACTOR FOR THE BUTTERWORTH FILTER IS GIVEN BY
C
C    GAIN=/H(W)/**2=1/(1+(TAN(W*T/2)/TAN(W1*T/2))**(2*N))
C
C HERE T IS THE SAMPLING INTERVAL, N IS COMPUTED BY SPECIFYING
C THE HALF POWER POINT W1 AND THE RELATIVE GAIN AT THE POINT
C W=W2>W1, AND THE TRANSFER FUNCTION H(Z) IS GIVEN BY THE TWO FORMULAS
C
C    H(Z)=SUM(1 TO N+1)(B(J)*Z**(-J+1))/(1+SUM(1 TO N)(A(J)*Z**(-J))
C
C    H(Z)=BETA*(1+Z)**N/(Z-P(1))*(Z-P(2)*...*(Z-P(N)).
```

84

```
C
C HERE THE POLES OF THE FILTER, P(I), ARE GIVEN BY
C
C    P(I)=(1-TAN(ARG)**2+(-1)**95*2*TAN(ARG)*SIN(TH(I))/
C         (1-2*TAN(ARG)*COS(TH(I))+TAN(ARG)**2)
C WHERE ARG=W1*T/2,TH(I)=(I-1)*PI/N FOR N ODD, =(2*I-1)*PI/2*N
C FOR N EVEN, AND ONLY THOSE N P(I) ARE CHOSEN THAT SATISFY
C /P(I)<1.. THE NORMALIZING FACTOR BETA IS GIVEN BY
C
C    BETA=(1-P(1))*(1-P(2))$...*(1-P(N))/2**N
C
C BY COMPARISON OF THE TWO EXPRESSIONS FOR THE TRANSFER FUNCTION
C THE FILTER COEFFICIENTS A(J) AND B(J) ARE DEDUCED
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      PI=3.1415927
C
C READ IN OR COMPUTE THE FREQUENCIES F1 AND F2, THE GAIN AT F2,GW28 AND
C THE SAMPLING INTERVAL
C
      TIME=0.001*XNYQ

      READ(NR,1) F1,F2,GW2
1     FORMAT(3E13.7)
      WRITE(NW,2) F1,F2,GW2,TIME
2     FORMAT(1H1,'   THE HALF POWER POINT IS ',E13.7,'MHZ',/,
     1'   THE GAIN AT ',E13.7,'MHZ IS',E13.7,/,
     2'    THE SAMPLING INTERVAL IS ',E13.7,'MICROSECONDS',/)
C
C COMPUTE AND PRINT OUT THE NUMBER OF POLES FOR THIS FILTER
C
      NMAX=10
      ARG1=F1*TIME*PI
      TAG1=TAN(ARG1)
```

```
        ARG2=F2*TIME*PI
        RT=(TAN(ARG2)/TAG1)**2
        TVAL=(1.0-GW2)/GW2
        N=1
        NOE=-1
 10     IF(RT**N.GE.TVAL) GO TO 20
        N=N+1
        NOE=-NOE
        IF(N.GT.NMAX) GO TO 25
        GO TO 10
 25     WRITE(NW,3) NMAX
 3      FORMAT(1X,' THE NUMBER OF POLES FOR THE FILTER EXCEEDED ',/,
       1'   THE ALLOWED LIMIT OF',I5)
        STOP
C
C COMPUTE ALL THE 2*N POLES WHICH ARE EITHER INSIDE OR OUTSIDE THE
C UNIT CIRCLE
C
 20     TWON=2*N
        ITWON=TWON
        EN=N
        WRITE(NW,4)
        WRITE(NW,5)
        DO 40 I=1,ITWON
        IF(NOE.EQ.1) GO TO 30
        TH=(I-1)*PI/EN
        GO TO 35
 30     TH=(2*I-1)*PI/TWON
 35     CTH(I)=COS(TH)
        STH(I)=SIN(TH)
 40     CONTINUE
        R1=1-TAG1**2
        R2=2*TAG1
        D1=1+TAG1**2
```

```
          K=0
          DO 50 I=1,ITWON
          DENOM=D1-R2*CTH(I)
          P(I)=R1/DENOM
          Q(I)=R2*STH(I)/DENOM
          PQ(I)=P(I)**2+Q(I)**2
          WRITE(NW,6) I,P(I),Q(I),PQ(I)
C
C SELECT ONLY THOSE POLES WHICH ARE INSIDE THE UNIT CIRCLE
C
          IF(PQ(I).GT.1.0) GO TO 50
          K=K+1
          ROOT(K)=CMPLX(P(I),Q(I))
50        CONTINUE
5         FORMAT(1H/,'INDEX',5X,'REAL PART',5X,'IMAG PART',8X,'SQUARE')
4         FORMAT(1H/,20X,'FILTER POLE LOCATIONS')
6         FORMAT(1X,I5,3(1X,E13.7))
          IF(K.EQ.N) GO TO 60
          WRITE(NW,7) K,ITWON
7         FORMAT(1H/,' ONLY',I3,' POLES OUT OF',I3,' ARE INSIDE UNIT CIRCLE'
     1,/,1X,'   PROGRAM EXECUTION HALTED')
          STOP
C
C COMPUTE BETA
C
60        BETA=CMPLX(1.0,0.0)
          UNIT=BETA
          DO 70 I=1,N
70        BETA=BETA*(UNIT-ROOT(I))/2.0
C
C COMPUTE THE BINOMIAL COEFFICIENTS
C
          INOM(1)=1
          NP1=N+1
```

```
      DO 75 I=2,NP1
75    INOM(I)=((N-I+2)*INOM(I-1))/(I-1)
      WRITE(NW,14) N
14    FORMAT(' BINOMIAL COEFFICIENTS FOR (1+Z)**',I2,/,
     1' INDEX COEFFICIENT')
      WRITE(NW,15) (I,INOM(I),I=1,NP1)
15    FORMAT(4X,I2,7X,I4)
C
C COMPUTE THE DIGITAL FILTER COEFFICIENTS
C
      IF(N-2)80,90,100
80    U(1)=-ROOT(1)
      GO TO 125
90    U(3)=CMPLX(1.0,0.0)
      U(2)=-ROOT(1)-ROOT(2)
      U(1)=ROOT(1)*ROOT(2)
      GO TO 125
100   U(1)=ROOT(1)*ROOT(2)
      U(2)=-ROOT(1)-ROOT(2)
      U(3)=CMPLX(1.0,0.0)
      L=3
      R(2)=CMPLX(1.0,0.0)
      DO 120 K=3,N
      R(1)=-ROOT(K)
      DO 110 I=1,L
110   S(I)=U(I)
      CALL POLMLT(2,R,L,S,LR,U)
      L=LR
120   CONTINUE
125   WRITE(NW,8)
8     FORMAT(1H/,' INDEX',5X,'REAL A(I)',5X,'IMAG A(I)')
      DO 130 I=1,N
      IS=N+1-I
      RA=REAL(U(IS))
```

```
          A(I)=RA
          AA=AIMAG(U(IS))
130       WRITE(NW,9) I,RA,AA
9         FORMAT(1X,I6,2(1X,E13.7))
          NP1=N+1
          WRITE(NW,11)
11        FORMAT(1H/,' INDEX',5X,'REAL B(I)',5X,'IMAG B(I)')
          NP2=N+2
          DO 140 I=1,NP1
          NR=NP2-I
          SHIT=INDM(NR)
          S(I)=CMPLX(SHIT,0.0)*BETA
          RB=REAL(S(I))
          B(I)=RB
          BB=AIMAG(S(I))
140       WRITE(NW,9) I,RB,BB
C
C DIGITALLY FILTER THE INPUT TIME SERIES
C
          DO 170 K=1,NSTAR
          Z=0.0
          DO 150 J=1,NP1
          IND=K-J+1
          IF(IND.LE.0) GO TO 155
          Z=Z+B(J)*AIMAG(YNYQA(IND))
150       CONTINUE
155       DO 160 J=1,N
          IND=K-J
          IF(IND.LE.0) GO TO 165
          Z=Z-A(J)*REAL(YNYQA(IND))
160       CONTINUE
165       X=AIMAG(YNYQA(K))
          YNYQA(K)=CMPLX(Z,X)
170       CONTINUE
```

```
       DO 180 K=1,NSTAR
       X=REAL(YNYQA(K))
180    YNYQA(K)=CMPLX(X,0.0)
       NS2=2*NSTAR
C
C PRINT OUT AND PLOT THE FILTERED TIME SERIES
C
       WRITE(NW,12)
       WRITE(NW,13) (ARRAY(I),I=1,NS2,2)
12     FORMAT(' FILTERED TIME SERIES')
13     FORMAT(8(1X,E13.7))
       RETURN
       END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE FILTER
    C         THIS IS A RECURSIVE FILTER ROUTINE WHERE NP IS THE NUMBER OF POLES, A
    C         AND B ARE THE FILTER COEFFICIENTS,
    C         AIMAG(YNYQA) IS THE INPUT ARRAY,
    C         REAL(YNYQA) IS THE OUTPUT ARRAY
    C         REAL(YNYQA(I))=SUMB(J)*AIMAG(YNYQA(I-J+1) - SUMA(J)*REAL(YNYQA(I-J)).
    C         FOR NO FILTERING SET B(1)=1,B(K)=0,K.NE.1;A(K)=0 ALL K.
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1   XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          DIMENSION ARRAY(4096)
          DIMENSION A(10),B(10)
          EQUIVALENCE (ARRAY(1),YNYQA(1))
          READ(NR,3) NP
    3     FORMAT(I5)
          READ(NR,4) (A(K),K=1,NP)
          NPL=NP+1
          READ(NR,4) (B(K),K=1,NPL)
    4     FORMAT(5(2X,E13.7))
          WRITE(NW,101)
    101   FORMAT(1H1,' FILTER COEFFICIENTS')
          WRITE(NW,102)  (K,A(K),K,B(K),K=1,NP)
    102   FORMAT(1X,'A(',I2,')=',E13.7,'B(',I2,')=',E13.7)
          WRITE(NW,103) NPL,B(NPL)
    103   FORMAT(21X,'B(',I2,')=',E13.7)
    C DIGITALLY FILTER THE INPUT TIMESERIES
    C
          DO 170 K=1,NSTAR
```

```
        Z=0.0
        DO 150 J=1,NP1
        IND=K-J+1
        IF(IND.LE.0) GO TO 155
        Z=Z+B(J)*AIMAG(YNYQA(IND))
150     CONTINUE
155     DO 160 J=1,NP
        IND=K-J
        IF(IND.LE.0) GO TO 165
        Z=Z-A(J)*REAL(YNYQA(IND))
160     CONTINUE
165     X=AIMAG(YNYQA(K))
        YNYQA(K)=CMPLX(Z,X)
170     CONTINUE
        DO 180 K=1,NSTAR
        X=REAL (YNYQA(K))
180     YNYQA(K)=CMPLX(X,0.0)
C
C PRINT OUT THE FILTERED TIME SERIES
C
        NS2=2*NSTAR
        WRITE(NW,12)
        WRITE(NW,13) (ARRAY (I),I=1,NS2,2)
12      FORMAT(30X,' FILTERED TIME SERIES',/)
13      FORMAT (8(1X,E13.7))
        RETURN
        END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                     SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE LNYQ
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1  XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          DIMENSION XOUT(4),YOUT(4)
    C THIS ROUTINE INTERPOLATES AT THE NYQUIST INTERVALS
    C USING A STRAIGHT LINE AS THE INTERPOLATING POLYNOMIAL
    C IF XF(I-1)<X<XF(I) THEN THE INTERPOLATED VALUE
    C IS Y WHERE Y=C1*X*C2 WITH
    C C1=(YF(I)-YF(I-1))/(XF(I)-XF(I-1))
    C C2=(YF(I-1)*XF(I)-YF(I)*XF(I-1))/(XF(I)-XF(I-1))
          IFLAG=-1
          L=1
          I=1
   20     IF(XF(I).GE.0)GO TO 30
          I=I+1
          IF(I-JF)20,150,150
   30     JST=I
          YNYQA(L)=CMPLX(0.0,YF(I))
   40     X=XF(JST)+L*XNYQ
   90     IF(X-XF(I))100,130,140
  100     IFLAG=IFLAG+1
  110     DENOM=XF(I)-XF(I-1)
          C1=(YF(I)-YF(I-1))/DENOM
          C2=(YF(I-1)*XF(I)-YF(I)*XF(I-1))/DENOM
  115     Y=C1*X+C2
  116     L=L+1
          IF(L-NSTAR)120,120,150
```

```
120     YNYQA(L)=CMPLX(0.0,Y)
        GO TO 40
130     Y=YF(I)
        GO TO 116
140     IFLAG=-1
        I=I+1
        IF(I-JF)90,90,145
145     I=JF
        GO TO 110
150     WRITE(NW,3)
        WRITE(NW,1)
        DO 160 K=1,NSTAR,4
        DO 155 J=1,4
        JP=J+K-1
        XOUT(J)=XF(JST)+(JP-1)*XNYQ
155     YOUT(J)=AIMAG(YNYQA(JP))
        WRITE(NW,2)((XOUT(L),YOUT(L)),L=1,4)
160     CONTINUE
2       FORMAT(8(2X,E13.7))
1       FORMAT(4(11X,'TIME',6X,'INTERP. Y'))
3       FORMAT(1H1,20X,'LISTING OF THE LINEARLY INTERPOLATED TIME SERIES')
        RETURN
        END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE NYQST
C NYQST TAKES  NON-EQUI SPACED ARRAY YF,XF AND PRODUCES AN EQUISPACED COMPLEX
C  ARRAY YNYQA.  INITIALLY RE(YNYQA) =0 AND IM(YNYQA)=INTERPOLATED VALUES
C  EQUISPACED ARRAY.
C YF(I), I=1,JF GIVEN AS V/DIV
C  XF(I) , I=1,JF GIVEN AS NS/DIV
C  XNYQ IS THE NYQUIST TIME INTERVAL IN NS/DIV
C  NSTAR = 2**NPOW IS GIVEN
C
C INITIALIZE VALUES
C
          DIMENSION XOUT(4),YOUT(4)
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,C1,ST,X0,Y0,
        1  XS,YS,T,V,XNYQ
        2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
        1IFPUND,IFPUVI,IFPUNR
          COMPLEX YNYQA
          DIMENSION A(4,4),B(4)
          LTSAV=0
          LBSAV=0
          ISAV=0
          L=1
          IST=1
C
C         SEARCH FOR FIRST XF > 0
C
20        IF(XF(IST).GE.0)GO TO 30
          IST=IST+1
          IF(IST - JF) 20,150,150
30        JSTRT=IST
```

```
          YNYQA(L)=CMPLX(0.0,YF(JSTRT))
          I=JSTRT
70        XVAL=XF(JSTRT) + L*XNYQ
75        IF(XF(I).GT.XVAL)GO TO 80
          I=I+1
          IF(I - JF) 75,115,75
80        LT=I+1
          LB=I-2
          IF(LB .LE. JSTRT)GO TO 110
          I=I-1
85        L=L+1
C
C         CHECK IF BOUNDS ARE SAME AS LAST TIME
C
          IF(LT.EQ.LTSAV .AND. LB.EQ.LBSAV .AND. I.EQ.ISAV)GO TO 100
C
C         NEW BOUNDS
C
          LTSAV=LT
          LBSAV=LB
          ISAV=I
          IF(L - NSTAR) 95,95,120
C
C         FIT THE FOUR POINTS TO A CUBIC
C
95        CALL LSQPOL(A,B,2,LB,LT)
          DENOM=A(1,1)*A(2,2)-A(1,2)**2
          T1=A(2,2)*B(1)-A(1,2)*B(2)
          T2=A(1,2)*B(1)-A(1,1)*B(2)
          B(1)=T1/DENOM
          B(2)=-T2/DENOM
100       YNYQA(L)=CMPLX(0.0,POLY(B,2,XVAL))
          IF(L-NSTAR)70,120,120
```

```
C      NEAR BEGINNING OF TRACE
C
110    LB=JSTRT
       LT=LB+3
       I=LB
       GO TO 85
C
C      NEAR END OF TRACE
C
115    LB=JF-3
       LT=JF
       I=I-1
       GO TO 85
150    WRITE(NW,1)
1      FORMAT(1X,' ALL TIMES ARE NEGATIVE')
       STOP
120    WRITE(NW,3)
       WRITE(NW,4)
       DO 200 K=1,NSTAR,4
       DO 190 J=1,4
       IND=K+J-1
       XOUT(J)=XF(JSTRT)+(IND-1)*XNYQ
190    YOUT(J)=AIMAG(YNYQA(IND))
       WRITE(NW,2)(((XOUT(L),YOUT(L)),L=1,4))
200    CONTINUE
2      FORMAT(8(2X,E13.7))
3      FORMAT(1H1,'LISTING OF LEAST SQUARES INTERPOLATED TIME SERIES')
4      FORMAT(4(11X,'TIME',6X,'INTERP. Y'))
       RETURN
       END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                    SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
        SUBROUTINE AUTCOR(K,NSAVE,NPSAV)
        COMMON YNYQA(2048)
        COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
       1  XS,YS,T,V,XNYQ
       2,NT,XZ,YZ,JD
        COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
       1IFPUND,IFPUNI,IFPUNR
        COMPLEX YNYQA
        DIMENSION FREQ(2048),POWER(2048),ZF(2048),OUT(8)
        DIMENSION ARRAY(4096)
        EQUIVALENCE (ARRAY(1),YNYQA(1))
        EQUIVALENCE (FREQ(1),ZF(1)),(POWER(1),XF(1))
        WRITE(NW,1)
     1  FORMAT('1 LISTING OF THE AUTOCORRELATION FUNCTION')
        PI=3.1415927
    C
    C COMPUTE AND PRINT OUT THE VALUES OF NSAVE,M,AND IH
    C
        KM1=K-1
        IH=1
        M=NSAVE-NSAVE*KM1/10
        NSTAR=M
        WRITE(NW,1001) NSAVE,IH,M
    C
    C COMPUTE DTAU=IH*XNYQ
    C
        DTAU=IH*XNYQ
 1001   FORMAT(' THERE ARE',I3,' VALUES IN THE DIGITAL TIME SERIES',/,
       1' THE LAG INTERVAL IS DTAU=IH*DT WHERE IH=',I3,/,'THERE ARE',I3,'
       2VALUES IN THE DIGITAL AUTOCORRELATION FUNCTION',/)
    C
    C COMPUTE THE AUTOCORRELATION FUNCTION AND STORE IT IN THE ARRAY XF
```

```
C
      DO 30 IR=1,M
      SUM=0.0
      INMHR=NSAVE-IH*(IR-1)
      DO 20 IQ=1,INMHR
      IMESS=IQ+IH*(IR-1)
20    SUM=SUM+YF(IQ)*YF(IMESS)
      SUM=SUM/INMHR
      XF(IR)=SUM
30    CONTINUE
C
C PRINT OUT AND PLOT THE AUTOCORRELATION FUNCTION
C
      WRITE(NW,5)
5     FORMAT(1X,4(10X,'TIME',7X,'AUTOCOR'))
      DO 35 I=1,M
35    YNYQA(I)=CMPLX(XF(I),0.0)
      XNYSV=XNYQ
      XNYQ=DTAU
      DO 37 I=1,M,4
      DO 36 J=1,4
      OUT(2*J-1)=XNYQ*(I-1+J-1)
36    OUT(2*J)=XF(I+J-1)
      WRITE(NW,4) (OUT(L),L=1,8)
69    FORMAT(4(7X,E13.7))
4     FORMAT(1X,8(1X,E13.7))
37    CONTINUE
      CALL PLOTS(4)
      XNYQ=XNYSV
C
C COMPUTE THE RAW SPECTRAL DENSITY ESTIMATES AND STORE THEM
C IN THE ARRAY ZF.
C
      DO 50 IR=1,M
```

```
        IPO=IR-1
        SUM=XF(1)+XF(M)*(-1.)**IPO
        MM1=M-1
        DO 40 IQ=2,MM1
40      SUM=SUM+2.0*XF(IQ)*COS(PI*(IQ-1)*(IR-1)/M)
        ZF(IR)=DTAU*SUM
50      CONTINUE
C
C COMPUTE THE REFINED SPECTRAL DENSITIES AND STORE THEM IN THE
C ARRAY XF
C
        XF(1)=0.54*ZF(1)+0.23*ZF(2)
        XF(M)=0.23*ZF(M-1)+0.54*ZF(M)
        DO 60 I=2,MM1
60      XF(I)=0.23*ZF(I-1)+0.54*ZF(I)+0.23*ZF(I+1)
        DO 70 I=1,M
70      ZF(I)=(I-1)/(2*M*DTAU)
C
C PRINT OUT AND PLOT THE REFINED SPECTRAL DENSITIES
C
        WRITE(NW,7)
7       FORMAT('1REFINED SPECTRAL DENSITY ESTIMATES FOR THE AUTOCORRELATIO
       1N FUNCTION')
        WRITE(NW,3)
3       FORMAT(1H/,4(5X,'FREQUENCY',9X,'POWER'),/)
        DO 80 I=1,M,4
80      WRITE(NW,4) (FREQ(I+J-1),POWER(I+J-1),J=1,4)
        NSTAR=2*M
        XNYSV=XNYQ
        XNYQ=DTAU
        DO 90 I=1,M
90      YNYQA(I)=CMPLX(POWER(I),0.0)
        CALL PLOTS(2)
        XNYQ=XNYSV
        NSTAR=NSAVE
        RETURN
        END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                    SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
            SUBROUTINE FFT
      C
      C THIS ROUTINE CALCULATES THE FOURIER TRANSFORM OF A NYQUIST SAMPLED ARRAY
      C THAT HAS ITS ELEMENTS IN BIT REVERSED ORDER. THE TECHNIQUE IS GENERALLY
      C KNOWN AS THE FAST FOURIER TRANSFORM.
      C
            COMMON YNYQA(2048)
            COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
           1 XS,YS,T,V,XNYQ
           2,VT,XZ,YZ,JD
            COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
           1IFPUND,IFPUNI,IFPUNR
            COMPLEX YNYQA,W,WE,FACTOR,SWAP,ZIP,S
            TPI=6.2831854
            DO 40 I=1,NPOW
            TI=2**I
            ARG=TPI/TI
            S=CMPLX(COS(ARG),SIN(ARG))
            IM1=I-1
            I1=2**IM1
            KSTOP=I1+1
            K=1
            W=CMPLX(1.0,0.0)
      20    SWAP=YNYQA(K)
            KPI1=K+I1
            FACTOR=YNYQA(KPI1)*W
            YNYQA(K)=SWAP+FACTOR
            YNYQA(KPI1)=SWAP-FACTOR
            K=K+1
            W=W*S
            IF(K-KSTOP)20,30,20
      30    K=KPI1+1
```

```
          KSTOP=K+IL
          W=CMPLX(1.0,0.0)
          IF(KSTOP.LE.NSTAR) GO TO 20
   40     CONTINUE
          DELF=1./(NSTAR*XNYQ)
          WRITE(NW,1)
          XSQ=XNYQ**2
          DO 50 I=1,NSTAR,ND
          ZIP=YNYQA(I)
          RZIP=REAL(ZIP)
          AIZIP=AIMAG(ZIP)
          FREQ=(I-1)*DELF
          PWR=(RZIP**2+AIZIP**2)*XSQ
          PHS=ATAN(AIZIP/RZIP)
          WRITE(NW,2)FREQ,RZIP,AIZIP,PWR,PHS
          YNYQA(I)=CMPLX(PWR,PHS)
   50     CONTINUE
   1      FORMAT(1H1,'     FREQUENCY     REAL F(W)      IMAG F(W)            POW
         1ER          PHASE')
   2      FORMAT(5(1X,E13.7))
          RETURN
          END
```

```
          SUBROUTINE FWAL
    C
    C THIS ROUTINE CALCULATES WALSH TRANSFORMS OF A NYQUIST
    C SAMPLED ARRAY. THE TECHNIQUE IS GENERALLY KNOWN AS THE FAST WALSH TRANSFORM.
    C
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1 XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          DO 10 I=1,NSTAR
          XF(I)=REAL(YNYQA(I))
    10    YF(I)=0.0
          DO 40 I=1,NPOW
          IM1=I-1
          I1=2**IM1
          KSTOP=I1+L
          L=1
          K=1
          J=0
          W=1.0
    20    S1=XF(K)
          KPI1=K+I1
          S2=XF(KPI1)
          SW2=S2*W
          INDEX=J/2
          SIG=(-1)**INDEX
          YF(L)=(S1+SW2)*SIG
          YF(L+1)=(-S1+SW2)*SIG
          K=K+1
```

```
        J=J+1
        L=L+2
        W=-W
        IF(K-KSTOP)20,30,20
30      K=KPI1+L
        KSTOP=K+I1
        J=0
        W=1
        IF(KSTOP.LE.NSTAR) GO TO 20
        DO 35 IZX=1,NSTAR
        XF(IZX)=YF(IZX)
35      YF(IZX)=0.0
40      CONTINUE
        DELF=1./(NSTAR*XNYQ)
        WRITE(NW,1)
        WRITE(NW,2)
        NSM1=NSTAR-1
        G=XF(1)**2
        L=0
        YNYQA(L+1)=CMPLX(G,0.0)
        WRITE(NW,3) L,XF(1),G
        DO 50 I=2,NSM1,2
        L=L+1
        AS=XF(I)
        AC=XF(I+1)
        G=AS**2+AC**2
        YNYQA(L+1)=CMPLX(G,0.0)
50      WRITE(NW,4) L,AS,AC,G
        L=L+1
        G=XF(NSTAR)**2
        YNYQA(L+1)=CMPLX(G,0.0)
        WRITE(NW,5) L,XF(NSTAR),G
1       FORMAT(1H1,' OUTPUT OF THE WALSH TRANSFORM')
2       FORMAT('       I',9X,'AS(I)',9X,'AC(I)',10X,'G(I)')
```

```
3        FORMAT(1X,I5,14X,2(1X,E13.7))
4        FORMAT(1X,I5,3(1X,E13.7))
5        FORMAT(1X,I5,1X,E13.7,15X,E13.7)
         RETURN
         END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                      SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE INVMAT(A,B,N,PIVOT,INV)
          DIMENSION A(4,4),B(4),INDEX(4,2),IPIVOT(4)
          EQUIVALENCE (JROW,IROW),(JCOLUM,ICOLUM),(AMAX,SWAP,T)
    C
    C
    C     MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF LINEAR EQUATIONS
    C
    C
    C     INITIALIZATION
    C
          DO 10 I=1,N
              IPIVOT(I)=0
    10        CONTINUE
          DO 100 I=1,N
    C
    C     SEARCH FOR PIVOT ELEMENT
    C
              AMAX=0.0
              DO 40 J=1,N
                  IF(IPIVOT(J).EQ.1) GO TO 40
                  DO 30 K=1,N
                      IF(IPIVOT(K)-1)20,30,130
    20                IF(ABS(AMAX).GT.ABS(A(J,K))) GO TO 30
                      IROW=J
                      ICOLUM=K
                      AMAX=A(J,K)
    30                CONTINUE
    40          CONTINUE
              IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1
    C
    C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
    C
```

```
                IF(IROW.EQ.ICOLUM) GO TO 60
                DO 50 J=1,N
                     SWAP=A(IROW,J)
                     A(IROW,J)=A(ICOLUM,J)
                     A(ICOLUM,J)=SWAP
     50              CONTINUE
                SWAP=B(IROW)
                B(IROW)=B(ICOLUM)
                B(ICOLUM)=SWAP
                PIVOT=A(ICOLUM,ICOLUM)
                INDEX(I,2)=ICOLUM
     60         INDEX(I,1)=IROW
                IF(PIVOT.EQ.0.0) GO TO 130
     C
     C
     C    DIVIDE PIVOT ROW BY PIVOT ELEMENT
     C
                A(ICOLUM,ICOLUM)=1.0
                DO 70 J=1,N
                     A(ICOLUM,J)=A(ICOLUM,J)/PIVOT
     70              CONTINUE
                B(ICOLUM)=B(ICOLUM)/PIVOT
     C
     C
     C    REDUCE NON-PIVOT ROWS
     C
                DO 90 J=1,N
                     IF(J.EQ.ICOLUM) GO TO 90
                     T=A(J,ICOLUM)
                     A(J,ICOLUM)=0.0
                     DO 80 K=1,N
                          A(J,K)=A(J,K)-A(ICOLUM,K)*T
     80                   CONTINUE
                     B(J)=B(J)-B(ICOLUM)*T
     90              CONTINUE
     100        CONTINUE
```

```
C
C       INTERCHANGE CCLUMNS
C
        DO 120 I=1,N
            J=N+1-I
            IF(INDEX(J,1).EQ.INDEX(J,2)) GO TO 120
            JROW=INDEX(J,1)
            JCOLUM=INDEX(J,2)
            DO 110 K=1,N
                SWAP=A(K,JROW)
                A(K,JROW)=A(K,JCOLUM)
                A(K,JCOLUM)=SWAP
110             CONTINUE
120         CONTINUE
130     RETURN
        END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                    SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE LSQPOL(A,B,N,LB,LT)
    C
    C THIS ROUTINE CALCULATES THE MATRIX A AND VECTOR B WHICH ARE USED TO SOLVE FOR
    C THE VECTOR X IN THE EQUATION A.X=B. THE VECTOR X HAS ELEMENTS WHICH CORRESPON
    C TO THE POLYNOMIAL COEFFICIENTS IN SUMCOEFF(I)*Z**I.THE COEFFICIENTS ARE
    C DETERMINED IN A LEAST SQUARES SOLUTION OF THE POLYNOMIAL TO SOME DATA.
    C
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1  XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMPLEX YNYQA
          DIMENSION A(4,4),B(4),C(10)
          LQ=2*N-1
          NPTS=LT-LB+1
          DO 10 I=1,N
          B(I)=0.0
          DO 10 J=1,N
    10    A(I,J)=0.0
          DO 5 L=1,LQ
    5     C(L)=0.0
          C(1)=NPTS
          DO 15 I=LB,LT
          DO 13 K=1,N
          KM1=K-1
          IF(XF(I).EQ.0.0.AND.KM1.EQ.0) GO TO 113
          B(K)=B(K)+YF(I)*(XF(I)**KM1)
          GO TO 13
    113   B(K)=B(K)+YF(I)
    13    CONTINUE
          DO 14 L=2,LQ
          LM1=L-1
```

```
14      C(L)=C(L)+XF(I)**LM1
15      CONTINUE
        DO 20 K=1,N
        KM1=K-1
        DO 20 KP=K,N
        A(K,KP)=C(KP+KM1)
20      A(KP,K)=A(K,KP)
        RETURN
        END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                  SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE PLCTS(N)
     C***  PLOT ONLY THE LOG OF THE POWER FOR THE MOMENT
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,X0,Y0,
         1  XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          DIMENSION ARR(126),PHI(1024),Y(1024)
          DATA BLANK,STAR,ZERO,PLUS,BAD/1H ,1H*,1H0,1H+,1HB/
          NCHAN=0
          GO TO (100,200,300,400),N
     100     DO 110 I=1,NSTAR,JD
          NCHAN=NCHAN+1
          PHI(NCHAN)=AIMAG(YNYQA(I))
     110    Y(NCHAN)=PHI(NCHAN)
          GO TO 500
     200    N1=NSTAR/2
          DO 210 I=1,N1,ND
          NCHAN=NCHAN+1
          PHI(NCHAN)=REAL(YNYQA(I))
     210    Y(NCHAN)=PHI(NCHAN)
          GO TO 500
     300    DO 310 I=1,NSTAR,ND .
          NCHAN=NCHAN+1
          PHI(NCHAN)=AIMAG(YNYQA(I))
     310    Y(NCHAN)=PHI(NCHAN)
          GO TO 500
     400     DO 410 I=1,NSTAR,JD
          NCHAN=NCHAN+1
          PHI(NCHAN)=REAL(YNYQA(I))
```

```
410     Y(NCHAN)=PHI(NCHAN)
500     NPSTEP=1
C***    THIS ROUTINE DOES THE PLOTTING OF THE DATA AND THEORETICAL FIT.
        WRITE(NW,1)
C***    INITIALIZE FOR FINDING MAXIMUM AND MINIMUM.
        AMAX=PHI(1)
        AMIN=AMAX
C***    INITIALIZE THE PLOTTING ARRAY TO BLANKS.
        DO 10 I=1,126
             ARR(I)=BLANK
10           CONTINUE
C***    FIND MAXIMUM AND MINIMUM OF EITHER THE DATA OR THE FIT.
        DO 30 I=1,NCHAN,NPSTEP
             IF(Y(I).GT.AMAX)    AMAX=Y(I)
             IF(Y(I).LT.AMIN)    AMIN=Y(I)
20           IF(PHI(I).GT.AMAX)    AMAX=PHI(I)
             IF(PHI(I).LT.AMIN)    AMIN=PHI(I)
30           CONTINUE
C***    FIND THE BIN SIZE.
        DBIN=(AMAX-AMIN)/124.0
C***    FIND WHERE TO PUT THE CURVES.
        DO 80 I=1,NCHAN,NPSTEP
             II=IFIX((PHI(I)-AMIN)/DBIN)+2
40           JJ=IFIX((Y(I)-AMIN)/DBIN)+2
             IF(II.NE.JJ) GO TO 50
             ARR(II)=PLUS
             GO TO 605
50           ARR(JJ)=ZERO
60           ARR(II)=STAR
605     GO TO (61,62,62,61),N
61      K=(I-1)*JD
        GO TO 70
62      K=(I-1)*ND
70      WRITE(NW,2) K,(ARR(K),K=1,126)
```

```
              ARR(II)=BLANK
              ARR(JJ)=BLANK
      80      CONTINUE
          DELF=1./(NSTAR*XNYQ)
          GO TO (81,82,83,84),N
      81      WRITE(NW,4)
      4       FORMAT(' PLOT OF AMPLITUDE VS TIME  AFTER INTERPOLATION')
              WRITE(NW,5) XNYQ
      5       FORMAT(' TO OBTAIN TIME VALUES MULTPLY XCOORD BY',E13.7)
              RETURN
      84      WRITE(NW,6)
      6       FORMAT(' PLOT OF THE FILTERED TIME SERIES')
              WRITE(NW,5) XNYQ
              RETURN
      82      WRITE(NW,7)
      7       FORMAT(' PLOT OF THE POWER VS. FREQUENCY')
              WRITE(NW,3) DELF
              RETURN
      83      WRITE(NW,8)
      8       FORMAT(' PLOT OF THE PHASE VS. FREQUENCY')
              WRITE(NW,3) DELF
              RETURN
      3       FORMAT(' TO OBTAIN FREQUENCY VALUES MULTIPLY X VALUE BY',E13.7)
      1       FORMAT (1H1)
      2       FORMAT (1X,I4,2X,126A1)
              END
```

COMPILER OPTIONS - NAME=   MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                    SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
```
          SUBROUTINE CSTOUT
    C     CHECK TIME ORDERING OF THE POINTS AND CAST OUT THOSE POINTS NOT IN THE
    C     PROPER TIME  ORDER
    C     IT ALSO AVERAGES THOSE POINTS WHICH HAVE THE SAME X VALUES
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1  XS,YS,T,V,XNYQ
         2,VT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          LSUM=1
          J=1
          JBAD=0
          K=2
          YSUM=YF(1)
          XSAV=XF(1)
    80    IF(XF(K)-XSAV)90,100,110
    90    K=K+1
          JBAD=JBAD + 1
          IF(K-JF)80,80,110
    100   LSUM=LSUM+1
          YSUM=YSUM+YF(K)
          K=K+1
          IF(K-JF)80,80,30
    110   YF(J)=YSUM/LSUM
          XF(J)=XSAV
          IF(K-JF)120,130,30
    120   XSAV=XF(K)
          YSUM=YF(K)
          LSUM=1
          K=K+1
```

```
        J=J+1
        GO TO 80
130     J=J+1
        YF(J)=YF(K)
        XF(J)=XF(K)
30      WRITE(NW,1) JBAD,JF,J
1       FORMAT(/,1X,I5,' POINTS OUT OF A TOTAL OF ',I5,
       1 ' DID NOT SATISFY THE TIME ORDER CRITERION',
       2 /,I5,' POINTS WILL BE USED')
        JF=J
        RETURN
        END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                  SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE SAVE (N)
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1  XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          IF(N) 10,10,20
   10     DO 15 I=1,NSTAR
   15     YF(I)=AIMAG (YNYQA(I))
          RETURN
   20     DO 25 I=1,NSTAR
   25     YF(I)=REAL (YNYQA(I))
          RETURN
          END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                  SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE UNSAVE(N)
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1   XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          IF (N) 10,10,20
   10     DO 15 I=1,NSTAR
   15     YNYQA(I)=CMPLX(0.0,YF(I))
          RETURN
   20     DO 25 I=1,NSTAR
   25     YNYQA(I)=CMPLX(YF(I),0.0)
          RETURN
          END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE ROT
C         ROTATE ZERO POINT OF THE TRACE
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1  XS,YS,T,V,XNYQ
         2,VT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          XZS=XZ-XO
          YZS=YZ-YO
          XZ= CT*XZS + ST*YZS
          YZ= -ST*XZS + CT*YZS
C         ROTATE AND SCALE INPUT ARRAY
          DO 10 I=1,JF
          SWX=XF(I) - XO
          SWY= YF(I) - YO
          YF(I)= ( -ST*SWX+CT*SWY - YZ)*V
   10     XF(I)=(CT*SWX+ST*SWY-XZ)*T
          RETURN
          END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                  SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE SCARTP
          COMMON YNYQA(2048)
          COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
         1  XS,YS,T,V,XNYQ
         2,NT,XZ,YZ,JD
          COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
         1IFPUND,IFPUNI,IFPUNR
          COMPLEX YNYQA
          DIMENSION XT(4),YT(4)
          READ(NR,2)XZ,YZ
          READ(NR,1) X1,X2,Y1,Y2
        1 FORMAT(4I5)
          READ(NR,2) (XT(I),YT(I),I=1,4)
        2 FORMAT(2(1X,E11.4))
          READ(NR,2)T,V
        3 FORMAT(1X,'THETA=',E13.7/1X,'COS(THETA)=',E13.7/
         X 1X,'SIN(THETA)=',E13.7/1X,'COORDINATE ORIGIN S=',E13.7/
         X 1X,'COORDINATE ORIGIN Y = ',E13.7,1X,'SCALE FACTOR X = ',E13.7/
         X 1X,'SCALE FACTOR Y = ',E13.7/)
        4 FORMAT(1X,'XAXIS PTS    X,Y COORDINATES',/,
         X 3(1X,E13.7),/,3(1X,E13.7))
        5 FORMAT(1X,'YAXIS PTS    X,Y COORDINATES',/,
         X 3(1X,E13.7),/,3(1X,E13.7))
          XO=ABS (X2*XT(1)-X1*XT(2))/ABS(Y1-X2)
          YO=ABS (Y2*YT(3)-Y1*YT(4))/ABS(Y1-Y2)
          XS=SQRT((XT(2)-XT(1))**2+(YT(2)-YT(1))**2)/ABS (X2-X1)
          YS=SQRT((XT(4)-XT(3))**2+(YT(4)-YT(3))**2)/ABS (Y2-Y1)
          IF(YT(2).EQ.YT(1)) GO TO 20
          TANT=((YT(2)-YT(1))/(XT(2)-XT(1))-(XT(4)-XT(3))/(YT(4)-YT(3)))/2.0
          CT=1.0/SQRT(1.0+TANT**2)
          ST=TANT*CT
          THETA=ATAN(TANT)
```

```
          GO TO 30
20        THETA=0.0
          ST=0.0
          CT=1.0
30        T=T/XS
          V=V/YS
          WRITE(NW,3) THETA,CT,ST,X0,Y0,XS,YS
          WRITE(NW,4) X1,XT(1),YT(2),X2,XT(2),YT(2)
          WRITE(NW,5) Y1,XT(3),YT(3),Y2,XT(4),YT(4)
          RETURN
          END
```

```
COMPILER OPTIONS - NAME=   MAIN,OPT=O2,LINECNT=57,SIZE=OOOOK,
                SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
           SUBROUTINE SRTFUR
      C
      C     THIS ROUTINE PERFORMS A BIT REVERSAL OF THE ORIGINAL NYQUIST ARRAY.
      C     IF I IS THE INDEX OF AN ELEMENT OF THE ARRAY THEN I=SUM(A(N)*2**N)
      C     WHERE THE LIMITS OF THE SUM ARE O AND M AND A(N) IS O OR 1.
      C     THE ELEMENT SPECIFIED BY I IS EXCHANGED WITH THE ELEMENT SPECIFIED
      C     BY J, WHERE J=SUM(A(M-N)*2**N.
      C     EXAMPLE: SUPPOSE I=O110101, THEN J=1010110.
      C     NSTAR=2**NPOW
            COMMON YNYQA(2048)
            COMMON NW,NR,XF(2048),YF(2048),NSTAR,NPOW,ND,JF,CT,ST,XO,YO,
           1  XS,YS,T,V,XNYQ
           2,NT,XZ,YZ,JD
            COMMON INTERP,IFFFT,IFFWAL,IFAUTO,IFGOF,IFPUNF,IFPUNW,IFPUNA,
           1IFPUND,IFPUNI,IFPUNR
            COMPLEX YNYQA,SWAP
            DIMENSION INT(15)
            DO 10 I=1,NPOW
      10    INT(I)=2**(NPOW-I)
            DO 30 I=1,NSTAR
            J=I-1
            ISUM=O
            DO 20 K=1,NPOW
            ISUM=ISUM + MOD(J,2)*INT(K)
      20    J=J/2
            ISUM=ISUM+1
            IF(ISUM-I)30,30,25
      25    SWAP=YNYQA(I)
            YNYQA(I)=YNYQA(ISUM)
            YNYQA(ISUM)=SWAP
      30    CONTINUE
            RETURN
            END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                  SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
        FUNCTION POLY(B,N,XVAL)
  C
  C EVALUATES A POLYNOMIAL FUNCTION
  C
        DIMENSION B(4)
        POLY=0.0
        XP=1.0
        DO 10 I=1,N
        POLY=POLY + B(I)*XP
  10    XP=XP*XVAL
        RETURN
        END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                  SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE POLMLT(M,A,N,B,L,C)
          COMPLEX C(40),B(20),A(20)
          L=M+N-1
          DO 10 I=1,L
   10     C(I)=CMPLX(0.0,0.0)
          DO 20 J=1,M
          DO 20 K=1,N
          I=J+K-1
   20     C(I)=C(I)+A(J)*B(K)
          RETURN
          END
```
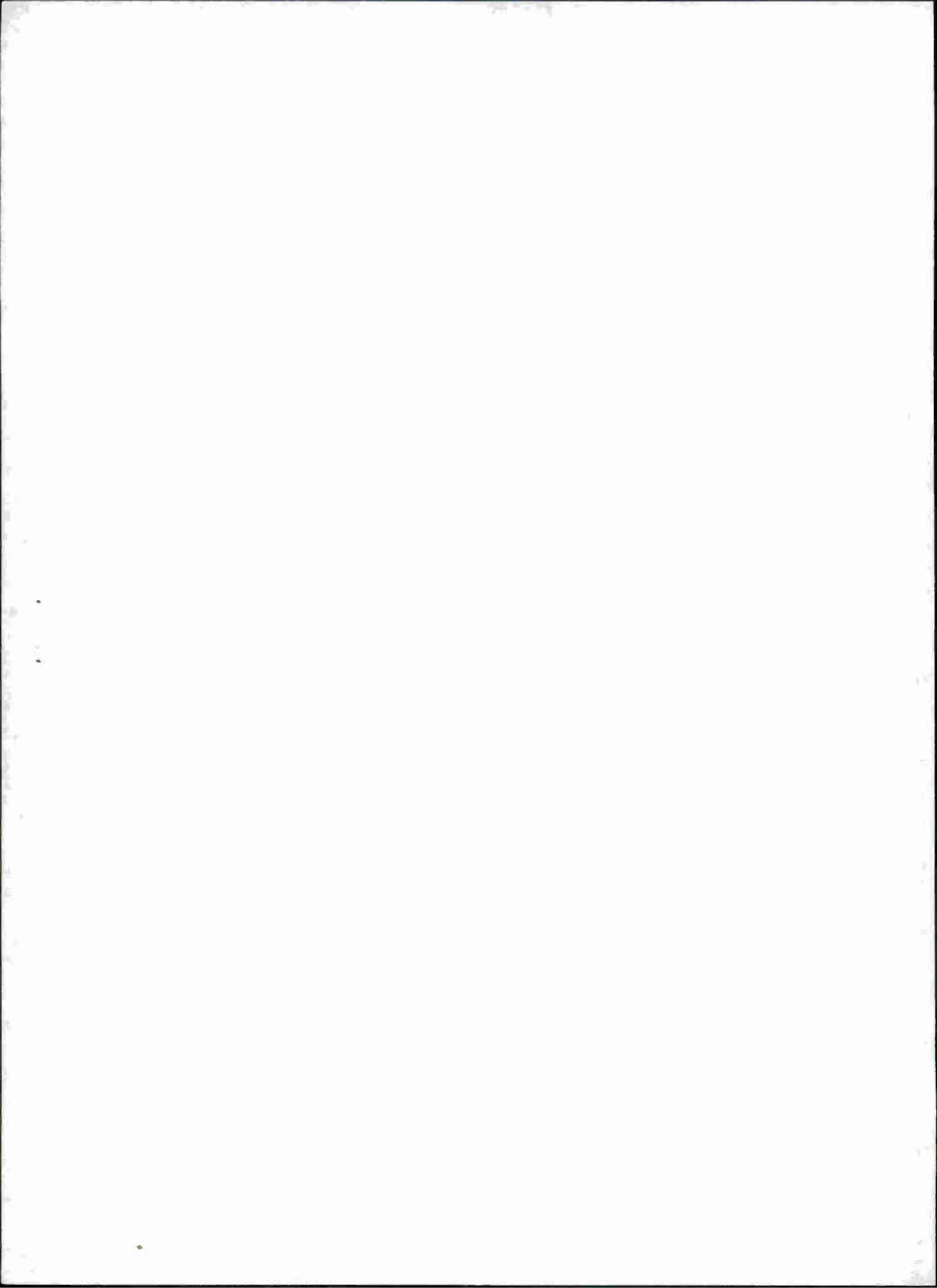
APPENDIX A

```
COMPILER OPTIONS - NAME=   MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
      C
      C MAIN PROGRAM FOR A GENERAL NON-LINEAR LEAST SQUARES FIT OF A
      C PARAMETRIZED FUNCTION TO A SET OF DATA Y
      C
      C
      C
      C      NON LINEAR LEAST SQUARES FITTING PROGRAM
      C
      C
      C
      C
      C************COMMON INFORMATION*********************************************
      C      A         MATRIX OF EQN. 8 AND INVERSE AFTER INVERSION
      C      C         VECTOR IN EQN.8
      C      DPR       CHANGES IN THE RESTRICTED PARAMETERS
      C      FMT       FORMAT FOR THE EXPERIMENTAL DATA
      C      PR        RESTRICTED PARAMETERS
      C      PSAVR     ARRAY FOR TEMPORARY SAVING OF PR
      C      Z         ARRAY OF DERIVATIVES
      C      PHI       VALUE OF THEORETICAL FUNCTION
      C      RESID     RESIDUALS
      C      X     ARRAY OF INDEPENDENT VARIABLES SUCH AS  TIME
      C      Y         THE EXPERIMENTAL DATA
      C      W         THE STATISTICAL WEIGHTS
      C      IFCOR     DETERMINES DISPOSITION OF CORRELATION COEFFICIENTS
      C      IFPLOT    DETERMINES IF THE PLOTTING PROGRAM IS CALLED
      C      INDEZ     INTERNAL PARAMETER COMMUNICATING BETWEEN MAIN AND TESTS
      C                THE INFORMATION THAT THE PROGRAM HAS OR HAS NOT
      C                CONVERGED
```

```
C      N3        THE NUMBER OF THE BEGINNING PARAMETER IN A GIVEN
C                REGION OF PARAMETER SPACE
C      N3ADPT    THE NUMBER OF BAD DATA POINTS
C      NCHAN     THE NUMBER OF CHANNELS
C      NE        THE NUMBER OF THE LAST PARAMETER IN A GIVEN REGION OF
C                PARAMETER SPACE
C      NP        NUMBER OF PARAMETERS TO BE FIT IN A GIVEN REGION OF
C                PARAMETER SPACE
C      NPSTEP    USED FOR PLOTTING
C      VP        THE NUMBER OF PARAMETERS TO BE FIT IN A GINEN REGION
C                OF PARAMETER SPACE
C      NRP       THE NUMBER OF RESTRICTED PARAMETERS
C      AA        ALPHA IN EQN. 12
C      CHI       CHI-SQUARED
C      CHIB      CHI-SQUARED BEFORE EACH ITERATION
C      CHIA      CHI-SQUARED AFTER EACH ITERATION
C      DEGF      THE NUMBER OF DEGREES OF FREEDOM
C*****************************************************************************
C
C
C***   I/O INFORMATION
C***   NREAD IS THE SYMBOLIC DESIGNATION FOR THE CARD READER
C***   NWRITE IS THE SYMBOLIC DESIGNATION FOR THE LINE PRINTER
C***   NPUNCH IS THE SYMBOLIC DESIGNATION FOR THE CARD PUNCH
C***   THE I/O CHANNELS ARE DEFINED IN THE NEXT THREE STATEMENTS
       COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
       COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
       COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
      1        VREAD,NWRITE,NPUNCH
       COMMON AA,CHI,CHIB,CHIA,DEGF
       NREAD=5
       NWRITE=6
       NPUNCH=7
```

```
C
C GET THE MAXIMUM # OF ITERATIONS AND THE INPUT DATA FORMAT
C
      READ(NREAD,1) JSTOP,FMT
C
C GET THE INPUT DATA FOR THE N-TH FIT
C
10    CALL READIN
C
C INITIALIZE ITERATION COUNTER
C
      JQUIT=0
      WRITE(NWRITE,2)
C
C INCREMENT ITERATION COUNTER
C
20    JQUIT=JQUIT+1
C
C TEST # OF ITERATIONS
C
      IF(JQUIT.LE.JSTOP)GO TO 30
      WRITE(NWRITE,3)
      GO TO 40
C
C SET UP MATRIX OF DERIVATES AND SOLVE FOR DELTA PARAMETERS
C
30    CALL GRIND(0)
C
C TEST AND MAKE CHANGES IN THE PARAMETERS
C
      CALL TESTS
C
C INDEZ=0 IF THE FIT HAS CONVERGED
```

```
C
        IF(INDEZ)20,40,40
C
C WRITE OUT THE PARAMETERS AND THE THEORETICAL FIT
C
40      CALL SCRIBE
C
C TEST IF A PLOT IS DESIRED
C
        IF(IFPLOT)60,70,60
C
C DO THE PLOTTING
C
60      CALL PLOTS
C
C DO THE ERROR ANALYSIS
C
70      CALL ERRMAT
C
C GO GET THE DATA FOR THE NEXT FIT
C
        IF(1)80,80,10
1       FORMAT(I5,5A4)
2       FORMAT(1H-,'CHANGES IN PARAMETERS')
3       FORMAT(1H-,'   THE NUMBER OF ITERATIONS EXCEEDED JSTOP.  THE',/,
     1          ' PROGRAM WILL AUTOMATICALLY GO TO THE PLOTTING AND ',/,
     2          ' ERROR ANALYSIS ROUTINES IN WHICH CAAE THE ANSWERS DO',/,
     3          ' NOT REPRESENT THE TRUE CHI-SQUARE SOLUTUON')
80      STOP
        END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE LSQPHI
      C
      C THIS ROUTINE CALCULATES THE THEORETICAL FUNCTION, RESIDUALS,
      C AND CHI-SQUARE. THE USER MUST SUPPLYTHE FUNCTION PHIFNC
      C
          COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
          COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
          COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
         1          NREAD,NWRITE,NPUNCH
          COMMON AA,CHI,CHIB,CHIA,DEGF
      C
      C CALCULATE CHI,PHI,AND RESID
      C
          CHI=0.0
          DO 10 I=1,NCHAN
          PHI(I)=PHIFNC(I)
          RESID(I)=(Y(I)-PHI(I))*W(I)
          CHI=CHI+RESID(I)**2
  10      CONTINUE
      C
      C DIVIDE CHI BY THE # OF DEGREES OF FREEDOM
      C
          CHI=CHI/DEGF
          RETURN
          END
```

129

```
COMPILER OPTIONS - NAME=   MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                  SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE ELTS(I)
      C
      C THIS ROUTINE CALCULATES THE DERIVATIVES OF PHI(I).
      C THE USER MUST SUPPLY THE FUNCTION DPHI(I,J)
      C
          COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
          COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
          COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
         1        NREAD,NWRITE,NPUNCH
          COMMON AA,CHI,CHIB,CHIA,DEGF
          DO 10 J=NB,NE
          Z(J)=DPHI(I,J)
   10     CONTINUE
          RETURN
          END
```

COMPILER OPTIONS - NAME=   MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                    SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF

```
          SUBROUTINE READIN
      C
      C***   THIS ROUTINE READS IN ALL THE INPUT PARAMETERS, DATA, AND SETS UP
      C***   INTERNAL CONTROL PARAMETERS.
      C
            COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
            COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
            COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
           1       NREAD,NWRITE,NPUNCH
            COMMON AA,CHI,CHIB,CHIA,DEGF
            DIMENSION NBAD(1024),YRUN(1024),TITLE(20)
            EQUIVALENCE (X(1),YRUN(1),NBAD(1)),(W(1),TITLE(1))
            WRITE(NWRITE,1001)
            READ(NREAD,1)TITLE
            WRITE(NWRITE,2) TITLE
      C
      C***   READ AND WRITE THE CONTROL PARAMETERS.
      C
            READ(NREAD,3) NCHAN,NBADPT,NRUNS,NRP,NPSTEP,IFPLOT,IFCOR
            WRITE(NWRITE,1002)
            WRITE(NWRITE,4) NCHAN,NBADPT,NRUNS,NRP,NPSTEP,IFPLOT,IFCOR
      C
      C***   CALCULATE THE INTERNAL CONTROL PARAMETERS.
      C
            IF(NPSTEP.EQ.0)  NPSTEP=1
            NPARAM=NRP
            NP=NRP
            NB=1
            NE=NRP
```

```
C
C***   READ AND WRITE THE PARAMETERS FOR THE FUNCTION.
C
       READ(NREAD,5)(PR(I),I=1,NRP)
       WRITE(NWRITE,1003)
       WRITE(NWRITE,6)(I,PR(I),I=1,NRP)
       DO 40 I=1,NRP
       P(I)=PR(I)
40     CONTINUE
C
C***   INITIALIZE FOR SUMMING NRUNS OF DATA.
C
       DO 130 I=1,NCHAN
              Y(I)=0.0
130           CONTINUE
C***   GET DATA AND SUM IT.
       DO 150 K=1,NRUNS
              READ(NREAD,FMT)(YRUN(I),I=1,NCHAN)
              DO 140 I=1,NCHAN
                     Y(I)=Y(I)+YRUN(I)
140                  CONTINUE
150           CONTINUE
C
C***   GET THE X AND W VALUES FOR EACH DATA POINT
C
       READ(NREAD,FMT) (X(I),I=1,NSTAR)
       READ(NREAD,FMT) (W(I),I=1,NSTAR)
C
C***   TEST FOR BAD DATA POINTS.
C
180    IF(NBADPT)210,210,190
190    WRITE(NWRITE,1006)
```

```
C***   READ AND WRITE CHANNEL # OF BAD POINTS.
       READ(NREAD,9)(NBAD(I),I=1,NBADPT)
       WRITE(NWRITE,9)(NBAD(I),I=1,NBADPT)
C***   DISCARD BAD DATA POINTS.
       DO 200 I=1,NBADPT
               J=NBAD(I)
               Y(J)=0.0
               W(J)=0.0
200            CONTINUE
C***   WRITE DATA.
210    WRITE(NWRITE,1007)
       WRITE(NWRITE,12)(Y(I),I=1,NCHAN)
       WRITE(NWRITE,1008)
       WRITE(NWRITE,12) (X(I),I=1,NCHAN)
       WRITE(NWRITE,1009)
       WRITE(NWRITE,12) (W(I),I=1,NCHAN)
       WRITE(NWRITE,1001)
C
C***   CALCULATE THE # OF DEGREES OF FREEDOM FOR THE FIT.
C
       DEGF=NCHAN-NBADPT-NRP
C
C***   CALCULATE THE INITIAL THEORETICAL FUNCTION AND CHI-SQUARED.
C
       CALL LSQPHI
       CHIB=CHI
C
C***   SET CHI AFTER AN ITERATION TO 100.0 IN CASE PROGRAM CAN MAKE NO
C***   SUCCESSFUL ITERATIONS.
C
       CHIA=100.0
1001   FORMAT (1H1,/,1H-)
1      FORMAT (20A4)
```

```
2     FORMAT (20X,20A4)
3     FORMAT(7I5)
1002  FORMAT(1H-'   NCHAN  NBADPT   NRUNS      NRP  NPSTEP  IFPLOT   ',
     1'IFCOR')
4     FORMAT(1X,7I8)
5     FORMAT(8F10.4)
1003  FORMAT (1H-'PARAMETER NUMBER',5X,'PARAMETER')
6     FORMAT (14X,I3,4X,F10.2)
7     FORMAT(4(I5,F10.5))
9     FORMAT(14I5)
1006  FORMAT ('-CHANNEL NUMBER OF THE BAD DATA POINTS')
1007  FORMAT (1H1,/,1H-,30X,'EXPERIMENTAL DATA',//)
1008  FORMAT(1H1,30X,'X VALUES FO THE EXPERIMENTAL DATA')
1009  FORMAT(1H1,30X,' W VALUES FO THE EXPERIMENTAL DATA')
12    FORMAT(10(1X,E12.6))
      RETURN
      END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE TESTS
      C
      C***  THIS ROUTINE DOES THE TESTING OF THE CHANGES IN THE PARAMETERS
      C
          COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
          COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
          COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
         1        NREAD,NWRITE,NPUNCH
          COMMON AA,CHI,CHIB,CHIA,DEGF
          DIMENSION WED(4),ZED(4)
      C***  AND PERFORMS THE NECESSARY CHANGES.
      C***  INITIALIZE FOR SUMMING.
          DT=0.0
      C***  SUM AND SAVE THE PARAMETERS.
          DO 10 I=NB,NE
              DT=DT+DPR(I)*C(I)
              PSAVR(I)=PR(I)
      10        CONTINUE
      C***  TEST FOR DT < ZERO.
          IF(DT)20,40,40
      C***  CHANGE THE SIGN OF THE DELTA P'S IF DT< ZERO.
      20    DO 30 I=NB,NE
              DPR(I)=-DPR(I)
      30        CONTINUE
          DT=-DT
      C***  USE THE FULL LENGTH OF THE CHANGES VECTOR TO FIND A NEW CHI.
      40    AA=1.0
      C***  FIND THE NEW CHI.
          CALL CALS2
      C***  TEST FOR NEW CHI SMALLER THAN OLD, IF NOT GO TO STATEMENT 90.
          IF(CHIB.LE.CHI)  GO TO 90
      C***  TEST FOR THE CHANGE IN CHI LESS THAN TERMINATION VALUE.
```

135

APPENDIX B

```
50      IF((CHIB-CHI).GT.0.1E-08) GO TO 80
C***    REACHES HERE IF TERMINATED.
        INDEZ=0
C***    SET CHI AFTER ITERATION EQUAL TO CHI.
60      CHIA=CHI
C***    WRITE OUT CHANGES IN PARAMETERS, LENGTH OF VECTOR,CHI BEFORE,
C***    AND CHI AFTER.
        WRITE(NWRITE,1)(I,DPR(I),I=NB,NE)
        WRITE(NWRITE,2)AA,CHIB,CHIA
C***    SET CHI BEFORE TO CHI AFTER.
70      CHIB=CHIA
        RETURN
C***    REACHES HERE IF NOT CONVERGED YET AND GOES BACK TO WRITE OUT.
80      INDEZ=-1
        GO TO 60
C***    REACHES HERE IF FULL LENGTH OF VECTOR WILL NOT LOWER CHI AND
C***    TRIES THE PARABOLAS DESCRIBED IN WRITE UP.
90      S=CHI
        AA=0.5
        CALL CALS2
        ZED(1)=CHI
        WED(1)=AA
        N=1
        K=0
        AA=DT/(S-CHIB+2.0*DT)
100     K=K+1
C***    IF THE FACTOR MULTIPLYING THE CHANGES VECTOR IS TOO SMALL SKIP IT
C***    FOR THE CHI AFTER CALCULATION.
        IF(AA.LT.0.1E-01) GO TO 110
        CALL CALS2
        N=N+1
```

```
      ZED(N)=CHI
      WED(N)=AA
110   IF(K-2)120,130,150
120   AA=1.0/DT
      GO TO 100
130   ZED1=CHIB+S-2.0*ZED(1)
      IF(ZED1)140,150,140
140   AA=(S+3.0*CHIB-4.0*ZED(1))/(4.0*ZED1)
      GO TO 100
150   AA=0.5
      CHI=ZED(1)
      DO 160 I=1,N
            IF(CHI.GT.ZED(I))  GO TO 160
            CHI=ZED(I)
            AA=WED(I)
160         CONTINUE
C*** TEST THE SMALLEST VALUE GIVEN BY THE PARABOLAS TO SEE IF IT IS
C*** SMALLER THAN CHI BEFORE, IF NOT RESET PARAMETERS AND GO TO NEXT
C*** SUBSPACE.
      IF(CHIB.LE.CHI)  GO TO 170
      CALL CALS2
      GO TO 50
C*** RESET PARAMETERS.
170   DO 180 I=NB,NE
            PR(I)=PSAVR(I)
180         CONTINUE
      AA=0.0
      CALL CALS2
      INDEZ=0
190   WRITE(NWRITE,1001)
      WRITE(NWRITE,1)(I,DPR(I),I=NB,NE)
```

```
1      FORMAT (6(4H DP(,I2,2H)=,E12.5,2X))
2      FORMAT (4H AA=,F14.7,5X,5HCHIB=,E14.7,3X,5HCHIA=,E14.7,/)
1001   FORMAT ('-THE FOLLOWING DPR(I) PRODUCED A DIVERGENT STEP WHICH ',
      1         'COULD NOT BE FIXED')
       RETURN
       END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
        SUBROUTINE GRIND(INV)
    C
    C***  THIS ROUTINE SETS UP THE MATRIX OF DERIVATIVES, THE CONSTANTS IN
    C***  THE NORMAL EQUATIONS, THEN CALLS THE MATRIX INVERTER, AND FINALLY
    C***  SHIFTS THE CONSTANTS AND THE SOLUTION TO MATCH THE PARAMETERS.
    C
          COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
          COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
          COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
         1        NREAD,NWRITE,NPUNCH
          COMMON AA,CHI,CHIB,CHIA,DEGF
          DIMENSION B(50)
          EQUIVALENCE (DPR(1),B(1))
    C***  INITIALIZE FOR SUMMING.
          DO 20 I=1,NP
              DO 10 J=I,NP
                  A(I,J)=0.0
   10             CONTINUE
              B(I)=0.0
   20         CONTINUE
    C***  SET UP MATRIX AND CONSTANTS.
          DO 70 I=1,NCHAN
    C***       TEST FOR BAD DATA POINTS.
              IF(Y(I))40,70,40
   40         CALL ELTS(I)
              DO 60 J=1,NP
                  DO 50 K=J,NP
                      A(J,K)=A(J,K)+Z(J)*Z(K)
   50                 CONTINUE
                  B(J)=B(J)+Z(J)*RESID(I)
   60             CONTINUE
   70         CONTINUE
```

```
C***   GET LOWER HALF OF MATRIX (THE MATRIX IS SYMETERIC).
       DO 90 J=1,NP
             DO 80 K=J,NP
                   A(K,J)=A(J,K)
80                 CONTINUE
C                  SAVE THE CONSTANTS.
             C(J)=B(J)
90           CONTINUE
C***   CALL THE MATRIX INVERTER.
       CALL INVMAT (NP,INV,ZZZ)
120    .RETURN
       END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
        SUBROUTINE INVMAT (N,INV,PIVOT)
        COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
        COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
        COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
       1        NREAD,NWRITE,NPUNCH
        COMMON AA,CHI,CHIB,CHIA,DEGF
        DIMENSION B(50),INDC(50),INDR(50),IPIVOT(50)
        EQUIVALENCE (JROW,IROW),(JCOLUM,ICOLUM),(AMAX,SWAP,T)
        EQUIVALENCE (DPR(1),B(1))
C*** MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF LINEAR EQUATIONS.
C********NOTE.   PIVOT IS RETURNED AS ZERO IF MATRIX IS SINGULAR.
C********NOTE.    INV=1 IF THE INVERSE IS DESIRED.
C***    INITIALIZATION
        DO 10 I=1,N
            IPIVOT(I)=0
 10         CONTINUE
        DO 100 I=1,N
C***            SEARCH FOR PIVOT ELEMENT.
                AMAX=0.0
                DO 40 J=1,N
                    IF(IPIVOT(J).EQ.1) GO TO 40
                    DO 30 K=1,N
                        IF(IPIVOT(K)-1)20,30,130
 20                     IF(ABS(AMAX).GT.ABS(A(J,K))) GO TO 30
                        IROW=J
                        ICOLUM=K
                        AMAX=A(J,K)
 30                     CONTINUE
 40                 CONTINUE
                IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1
C***            INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL.
                IF(IROW.EQ.ICOLUM) GO TO 60
```

```
            DO 50 J=1,N
                    SWAP=A(IROW,J)
                    A(IROW,J)=A(ICOLUM,J)
                    A(ICOLUM,J)=SWAP
     50             CONTINUE
            SWAP=B(IROW)
            B(IROW)=B(ICOLUM)
            B(ICOLUM)=SWAP
     60     PIVOT=A(ICOLUM,ICOLUM)
     C***   TEST FOR SINGULAR MATRIX.
            IF(PIVOT.EQ.0.0) GO TO 130
     C***   DIVIDE PIVOT ROW BY PIVOT ELEMENT.
            DPIVOT=1.0/PIVOT
            A(ICOLUM,ICOLUM)=1.0
            DO 70 J=1,N
                    A(ICOLUM,J)=A(ICOLUM,J)*DPIVOT
     70             CONTINUE
            B(ICOLUM)=B(ICOLUM)*DPIVOT
     C***   REDUCE NON-PIVOT ROWS.
            DO 90 J=1,N
                    IF(J.EQ.ICOLUM) GO TO 90
                    T=A(J,ICOLUM)
                    A(J,ICOLUM)=0.0
                    DO 80 K=1,N
                            A(J,K)=A(J,K)-A(ICOLUM,K)*T
     80                     CONTINUE
                    B(J)=B(J)-B(ICOLUM)*T
     90             CONTINUE
     C***   SET INDEX IF INVERSE IS DESIRED.
            IF(INV.NE.1)  GO TO 100
            INDR(I)=IROW
            INDC(I)=ICOLUM
```

142

```
100          CONTINUE
C***   INTERCHANGE COLUMNS IF INVERSE IS DESIRED.
       IF(INV.NE.1)  GO TO 130
       DO 120 I=1,N
              J=N+1-I
              IF(INDR(J).EQ.INDC(J))  GO TO 120
              JROW=INDR(J)
              JCOLUM=INDC(J)
              DO 110 K=1,N
                     SWAP=A(K,JROW)
                     A(K,JROW)=A(K,JCOLUM)
                     A(K,JCOLUM)=SWAP
110                  CONTINUE
120           CONTINUE
130    RETURN
       END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
         SUBROUTINE CALS2
      C
      C***  THIS ROUTINE IS CALLED BY TESTS TO SET UP THE PARAMETERS
      C***  AFTER THE PERAMETERS HAVE BEEN CHANGED, AND CALCULATE CHI BY
      C***  BY CALLING LSQPHI
      C
         COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
         COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
         COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
        1       NREAD,NWRITE,NPUNCH
         COMMON AA,CHI,CHIB,CHIA,DEGF
      C***  FIND THE CHANGED RESTRICTED PARAMETERS.
         DO 10 I=NB,NE
             PR(I)=PSAVR(I)+AA*DPR(I)
         P(I)=PR(I)
      10         CONTINUE
      C
      C***  FIND THE CHI.
      C
      60    CALL LSQPHI
            RETURN
            END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE SCRIBE
      C
      C***   THIS ROUTINE DOES THE WRITING OUT.
      C
          COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
          COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
          COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
         1      NREAD,NWRITE,NPUNCH
          COMMON AA,CHI,CHIB,CHIA,DEGF
          DIMENSION PHIN(512),YN(512),XVEL(512)
          WRITE(NWRITE,1)
          WRITE(NWRITE,2)(I,PR(I),I=1,NRP)
       20 WRITE(NWRITE,3)
          WRITE(NWRITE,4)(PHI(I),I=1,NCHAN)
        1 FORMAT ('-PARAMETER NUMBER',10X,'PARAMETER')
        2 FORMAT (14X,I3,5X,E14.7)
        3 FORMAT (1H1,/,1H-,30X,'THEORETICAL FIT',//)
        4 FORMAT(8(2X,E13.7))
       30 RETURN
          END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                   SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE ERRMAT
    C
    C***   THIS ROUTINE DOES THE ERROR ANALYSIS.
    C
          COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
          COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
          COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
         1      NREAD,NWRITE,NPUNCH
          COMMON AA,CHI,CHIB,CHIA,DEGF
          DIMENSION ACSRS(50),AMS(50),FINT(50),SIGI(50)
    C***   TEST FOR NO SUCCESSFUL ITERATIONS.
          IF(CHIA.EQ.100.0)  GO TO 160
          PIE=3.1415927
    C***   INITIALIZE FOR FINDING THE DATA POINT WITH MAXIMUM WEIGHTED
    C***   RESIDUAL.
          TAX=0.0
    C***   FIND CHANNEL # OF MAXIMUM WEIGHTED RESIDUAL AND WRITE IT OUT.
          DO 10 I=1,NCHAN
                Q=ABS(RESID(I))
                IF(Q.LT.TAX)  GO TO 10
                ITRACK=I
                TAX=Q
    10          CONTINUE
          WRITE(NWRITE,1)ITRACK
    C***   FIND THE PROBABILITY FOR THE FITTED CHI.
          AR=SQRT(2.0*CHIA*DEGF)-SQRT(2.0*DEGF-1.0)
          PROB=ERFC(AR)/2.0
          NCH=NCHAN-NBADPT
    C***   WRITE OUT STATISTICAL INFORMATION.
          WRITE(NWRITE,2)NCH,NRP,CHIA,PROB
    C***   TEST CHI FOR LESS THAN 1.0 AND SET TO ONE IS SATISFIED.
          IF(CHIA.LT.1.0)  CHIA=1.0
```

```
C***    CALL GRIND TO FIND THE ERROR MATRIX.
20      CALL GRIND(1)
C***    FIND THE STANDARD DEVIATIONS ON THE FITTED PARAMETERS.
        DO 30 I=1,NRP
            ACSRS(I)=SQRT(A(I,I)*CHIA)
30          CONTINUE
C***    FIND THE CORRELATION COEFFICIENTS.
        II=NRP-1
        DO 50 I=1,II
            JJ=I+1
            SIG=ACSRS(I)
            DO 40 J=JJ,NRP
                A(I,J)=A(I,J)*CHIA/(SIG*ACSRS(J))
40              CONTINUE
50          CONTINUE
C***    WRITE OUT THE PARAMETERS AND THEIR STANDARD DEVIATIONS.
        WRITE(NWRITE,3)
        WRITE(NWRITE,4)(I,PR(I),ACSRS(I),I=1,NRP)
C***    TEST FOR CORRELATION COEFFICIENTS DESIRED.
100     IF(IFCOR)110,150,120
C***    IF IFCOR < ZERO PUNCH OUT PARAMETERS.
110     WRITE(NPUNCH,11)(PR(I),ACSRS(I),I=1,NRP)
120     WRITE(NWRITE,12)
C***    WRITE OUT CORRELATION COEFFICIENTS.
        II=NRP-1
        DO 140 I=1,II
            JJ=I+1
            WRITE(NWRITE,13)(I,J,A(I,J),J=JJ,NRP)
C***        TEST FOR PUNCHING.
            IF(IFCOR)130,140,140
130         WRITE(NPUNCH,14)(A(I,J),J=JJ,NRP)
```

```
140         CONTINUE
150    RETURN
160    WRITE(NWRITE,15)
1      FORMAT ('-CHANNEL NUMBER OF THE WORST DATA POINT IS ',I5)
2      FORMAT ('-CHI-SQUARED FOR',I5,' CHANNELS AND',I5,' PARAMETERS IS',
      1          F11.7,/,' THE PROBABILITY FOR WHICH IS',E16.7)
3      FORMAT (1H1,/,'-PARAMETER NUMBER',10X,'PARAMETER',10X,'STANDARD ',
      1          'DEVIATION')
4      FORMAT (14X,I3,5X,E14.7,14X,E14.7)
6      FORMAT (1X,E14.7,15X,E14.7)
11     FORMAT (5E16.7)
12     FORMAT (1H1,/,1H-,30X,'CORRELATION COEFFICENTS')
13     FORMAT (1H-,6(3H A(,I2,1H,,I2,2H)=,F10.7))
14     FORMAT (8F10.5)
15     FORMAT (' PROGRAM COULD MAKE NO SUCCESSFUL ITERATIONS')
       RETURN
       END
```

```
COMPILER OPTIONS - NAME=   MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                    SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          SUBROUTINE PLOTS
      C
      C***   THIS ROUTINE DOES THE PLOTTING OF THE DATA AND THEORETICAL FIT.
      C
          COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
          COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
          COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
         1         NREAD,NWRITE,NPUNCH
          COMMON AA,CHI,CHIB,CHIA,DEGF
          DIMENSION ARR(126)
          WRITE(NWRITE,1)
          DATA BLANK,STAR,ZERO,PLUS,BAD/1H ,1H*,1H0,1H+,1HB/
      C***   INITIALIZE FOR FINDING MAXIMUM AND MINIMUM.
          AMAX=PHI(1)
          AMIN=AMAX
      C***   INITIALIZE THE PLOTTING ARRAY TO BLANKS.
          DO 10 I=1,126
              ARR(I)=BLANK
      10          CONTINUE
      C***   FIND MAXIMUM AND MINIMUM OF EITHER THE DATA OR THE FIT.
          DO 30 I=1,NCHAN,NPSTEP
              IF(Y(I).EQ.0.0)  GO TO 20
              IF(Y(I).GT.AMAX)   AMAX=Y(I)
              IF(Y(I).LT.AMIN)   AMIN=Y(I)
      20          IF(PHI(I).GT.AMAX)   AMAX=PHI(I)
              IF(PHI(I).LT.AMIN)   AMIN=PHI(I)
      30          CONTINUE
      C***   FIND THE BIN SIZE.
          DBIN=(AMAX-AMIN)/124.0
      C***   FIND WHERE TO PUT THE CURVES.
          DO 80 I=1,NCHAN,NPSTEP
              II=IFIX((PHI(I)-AMIN)/DBIN)+2
```

```
                IF(Y(I).NE.0.0)   GO TO 40
                JJ=1
                ARR(1)=BAD
                GO TO 60
     40         JJ=IFIX((Y(I)-AMIN)/DBIN)+2
                IF(II.NE.JJ) GO TO 50
                ARR(II)=PLUS
                GO TO 70
     50         ARR(JJ)=ZERO
     60         ARR(II)=STAR
     70         WRITE(NWRITE,2)I,(ARR(K),K=1,126)
                ARR(II)=BLANK
                ARR(JJ)=BLANK
     80         CONTINUE
          WRITE(NWRITE,3)
     1    FORMAT (1H1)
     2    FORMAT (1X,I4,2X,126A1)
     3    FORMAT ('-0=EXPERIMENTAL POINT,  *=FITTED POINT,  +=COINCIDENT ',
          1        'EXPERIMENTAL AND FITTED POINTS,  B=BAD EXPERIMENTAL ',
          2        'POINT')
          RETURN
          END
```

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          FUNCTION PHIFNC(I)
          COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
          COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
          COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
         1        NREAD,NWRITE,NPUNCH
          COMMON AA,CHI,CHIB,CHIA,DEGF
          DIMENSION PSX(50)
      C
      C THIS SUBROUTINE CALCULATES  THE THEORETICAL FUNCTION FOR THE FITTING
      C PROCESS . SPECIFICALLY  PHI =SUN(P(K)*EXP(-P(K+1)*X(I))*COS(P(K+2)*X(I))
      C WHERE THE SUM IS OVER THE TOTAL NUMBER OF FUNCTIONS
      C
          XI=X(I)
          DO 10 K=1,NP
          PSX(K)=P(K)*XI
   10     CONTINUE
          PHIFNC=0.0
          DO 20 J=1,NP,3
   20     PHIFNC=PHIFNC+P(J)*EXP(-PSX(J+1))*COS(PSX(J+2))
          RETURN
          END
```

151

```
COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=57,SIZE=0000K,
                  SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,NOXREF
          FUNCTION DPHI(I,J)
          COMMON A(50,50),C(50),DPR(50),FMT(5),PR(50),PSAVR(50),Z(50),P(50)
          COMMON PHI(1024),RESID(1024),X(1024),Y(1024),W(1024)
          COMMON IFCOR,IFPLOT,INDEZ,NB,NBADPT,NCHAN,NE,NP,NPSTEP,NRP,
         1        NREAD,NWRITE,NPUNCH
          COMMON AA,CHI,CHIB,CHIA,DEGF
          DIMENSION PSX(50)
          DIMENSION SAVS(50),SAVC(50)
      C
      C THIS ROUTINE CALCULATES THE DERIVATIVES OF THE THEORETICAL FUNCTION
      C WHERE THE FUNCTION IS GIVEN BY
      C PHI = SUM(P(K)*EXP(-P(K+1)*X(I))*COS(P(K+2)*X(I))) WHERE THE
      C SUM IS OVER ALL FUNCTIONS
      C
          IF(J.NE.1) GO TO 20
          XI=X(I)
          DO 10 K=1,NP
          PSX(K)=P(K)*XI
   10     CONTINUE
          K=0
          DO 15 L=1,NP,3
          K=K+1
          CON=P(L)*EXP(-PSX(L+1))
          SAVC(K)=CON*COS(PSX(L+2))
          SAVS(K)=CON*SIN(PSX(L+2))
   15     CONTINUE
   20     ITE=J-1
          K=MOD(ITE,3)+1
          L=(J-1)/3+1
          GO TO (1,2,3),K
    1     DPHI=SAVC(L)/P(J)
          RETURN
```

```
2       DPHI=-XI*SAVC(L)
        RETURN
3       DPHI=-XI*SAVS(L)
        RETURN
        END
```
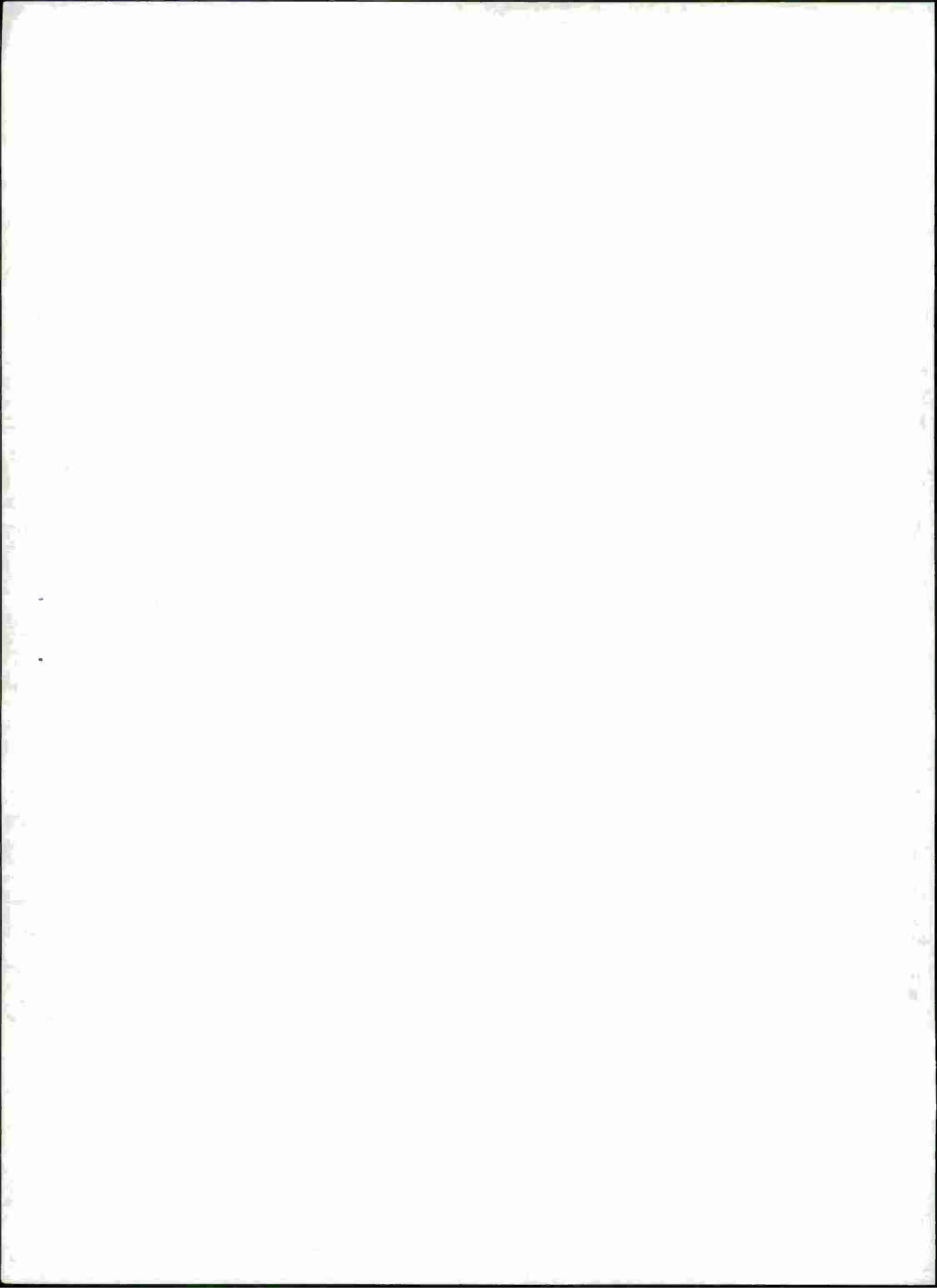
DISTRIBUTION

DEFENSE DOCUMENTATION CENTER
CAMERON STATION, BUILDING 5
ALEXANDRIA, VA 22314
  ATTN DDC-TCA 12 COPIES

OFC, CHIEF OF RESEARCH &
 DEVELOPMENT
USA RSCH & DEV GROUP (EUROPE)
BOX 15
FPR NEW YORK 09510
  ATTN LTC EDWARD E CHICK
       CHIEF, MATERIALS BRANCH

COMMANDER
US ARMY MATERIEL COMMAND
5001 EISENHOWER AVENUE
ALEXANDRIA, VA 22333
  ATTN AMCRD, RES, DEV & ENGR
       DIRECTORATE
  ATTN AMCRD-WN, JOHN CORRIGAN
  ATTN AMCMS-IS, ALDRIC SAUCIER

COMMANDER
USA ARMAMENTS COMMAND
ROCK ISLAND, IL 61201
  ATTN AMSAR-ASF, FUZE DIV
  ATTN AMSAR-RDF, SYS DEV DIV
       - FUZES

COMMANDER
USA MISSILE + MUNITIONS CENTER
 + SCHOOL
REDSTONE ARSENAL, AL 35809
  ATTN ATSK-CTD-P

DIRECTOR
DEFENSE ADVANCED RESEARCH PROJECTS
 AGENCY
ARCHITECT BUILDING
1400 WILSON BLVD
ARLINGTON, VA 22209
  ATTN DIR, TECH INFO, F. A. KOETHER

COMMANDER IN CHIEF
CONTINENTAL AIR DEFENSE COMMAND
ENT AFB
COLORADO SPRINGS, CO 80912
  ATTN DCS/C&E (CESA),
       DR. J. K. STERRETT
  ATTN CPPA, DIR. W. R. MATOUSH

ASSISTANT TO THE SECRETARY OF
 DEFENSE
TELECOMMUNICATIONS
WASHINGTON, DC 20301
  ATTN OPERATIONS & ENGINEERING

DEFENSE CIVIL PREPAREDNESS AGENCY
WASHINGTON, DC 20301
  ATTN TS(AED), RM 1C 535
  ATTN RE(SS), H. E. RODERICK
  ATTN C. VANDENBERGHE, RM 1E 542

COMMANDER
NATIONAL MILITARY COMMAND SYSTEM
 SUPPORT CENTER
WASHINGTON, DC 20305
  ATTTN CODE 350, J. A. KRECK
  ATTN CODE 340, W. H. DIX
  ATTN CODE 931
  ATTN CODE B210, WPN SYS ANAL DIV
  ATTN CODE 320.4
  ATTN CODE 950
  ATTN CODE 400
  ATTN CODE 470

NATIONAL COMMUNICATION SYSTEM
OFFICE OF THE MANAGER
WASHINGTON, DC 20305
  ATTN NCS-DE, DENNIS BODSON

DEFENSE COMMUNICATION ENGINEERING
 OFFICE
1860 WIEHLE AVENUE
RESTON, VA 22070
  ATTN H620
  ATTN R280
  ATTN R320
  ATTN R150
  ATTN R1035

DIRECTOR
DEFENSE INTELLIGENCE AGENCY
WASHINGTON, DC 20301
  ATTN DI-7B, PHYS VUL DIV,
       E. O. O'FARRELL

DIRECTOR
DEFENSE NUCLEAR AGENCY
WASHINGTON, DC 20305
  ATTN STRA, RADIATION DIRECTORATE
  ATTN RAEV, ELECTRONICS VULNERABILITY
       DIVISION
  ATTN APTL, DASA TECH LIBRARY (2 CY)
  ATTN STVL, VULNERABILITY DIRECTORATE
  ATTN PETER HAAS, DEP. DIR,
       SCIENTIFIC TECHNOLOGY
  ATTN APSI (ARCHIVES)
  ATTN VLIS (2 CY)

COMMANDER
FIELD COMMAND
DEFENSE NUCLEAR AGENCY
KIRTLAND AFB, NM 87115
  ATTN FCTA-A, TECHNICAL LIBRARY

COMMANDER
LIVERMORE DIVISION, FIELD COMMAND, DNA
LAWRENCE LIVERMORE LABORATORY
P.O. BOX 808
LIVERMORE, CA 94550
  ATTN DOCUMENT CONTROL

DIRECTOR OF DEFENSE RES AND
 ENGINEERING
WASHINGTON, DC 20301
  ATTN SPECIAL ASST(NET TECH ASSESS),
       N. F. WIKNER
  ATTN DEP DIR(STRATEGIC SYSTEMS)
  ATTN ASST DIR(ELECT & PHYS SCI),
       G. H. HEILMEIER
  ATTN ASST DIR STRAT & SUP SYS
       TEST AND EVAL
  ATTN EXEC ASST TO DIR
  ATTN DEP DIR STRAT & SPACE SYS

CHAIRMAN
OFFICE OF THE JOINT CHIEFS OF STAFF
WASHINGTON, DC 20301
  ATTN J-5,PLANS AND POLICY
       (R AND D DIVISION)
  ATTN J-6, CSD-1
  ATTN J6, LCOL LOGSDEN
  ATTN J-5, NUC/BIO/CHEM BRANCH
  ATTN J-32, WWMCCS PLANS DIV
  ATTN SAGA/OJCS, COL STEWART

DIRECTOR
JOINT STRAT TARGET PLANNING STAFF
OFFUTT AFB
OMAHA, NE 68113
  ATTN JLTW
  ATTN JPST

COMMANDER
NATIONAL MILITARY COMMAND SYS SUP CENTER
WASHINGTON, DC 20305
  ATTN B210, RM BE 685
  ATTN CODE 931
  ATTN CODE 430 (3 CY)

DIRECTOR
NATIONAL SECURITY AGENCY
FORT GEORGE G. MEADE, MD 20755
  ATTN O. O. VAN GUNTEN, R-425
  ATTN TDL
  ATTN SOG, KEN EDWARDS

HEADQUARTERS
EUROPEAN COMMAND, JCS
J-5
APO NEW YORK, NY 09128
  ATTN ECJE-WP

HEADQUARTERS
US EUROPEAN COMMAND
APO NEW YORK, NY 09055
  ATTN ECCE-PT

COMMANDER-IN-CHIEF
PACIFIC COMMAND
FPO SAN FRANCISCO, CA 96610
  ATTN J305

DIRECTOR
BALLISTIC MISSILE DEFENSE PROG OFFICE
1300 WILSON BLVD
ARLINGTON, VA 22209
  ATTN DOCUMENT CONTROL

DIRECTOR
WEAPONS SYSTEMS EVAL GROUP, ODDRE
OFFICE, SECRETARY OF DEFENSE
400 ARMY-NAVY DRIVE
WASHINGTON, DC 20305
  ATTN CAPT DONALD E. McCOY, USN
  ATTN H. S. KNAPP, JR.

OFFICE OF THE SECRETARY OF THE ARMY
WASHINGTON, DC 20310
  ATTN OUSA (OR), DR. DANIEL WILLIARD

ASSISTANT CHIEF OF STAFF FOR
COMMUNICATIONS-ELECTRONICS
DEPARTMENT OF THE ARMY
WASHINGTON, DC 20314
  ATTN CEED-7, WESLEY T. HEATH

ASSISTANT CHIEF OF STAFF FOR INTELLIGENCE
DEPARTMENT OF THE ARMY
WASHINGTON, DC 20310
  ATTN MITA/OM LIBRARY

COMMANDER-IN-CHIEF
USA EUROPE AND 7TH ARMY HQ
APO NEW YORK, NY 09403
  ATTN AEAGE-T2
  ATTN ODCSC-E, AEAGE-PI

155

OFFICE, CHIEF OF RESEARCH,
DEVELOPMENT & ACQUISITION
DEPARTMENT OF THE ARMY
WASHINGTON, DC 20310
  ATTN DAMA-MDO, LTC H. SLOAN
  ATTN DAMA-DDM-W, MAJ B. GRIGGS
  ATTN ABMDA-OO

COMMANDER
USA BALLISTIC MISSILE DEFENSE
  ADVANCED TECHNOLOGY CENTER
HUNTSVILLE OFFICE
P.O. BOX 1500
HUNTSVILLE, AL 35807
  ATTN RDMH-O
  ATTN RDMH-X
  ATTN CRDABH-S, ROLAND BROWN

COMMANDER
USA SAFEGUARD COMMUNICATIONS AGENCY
  ATTN DOCUMENT CONTROL
FORT HUACHUCA, AZ 85613
  ATTN SPECIAL SCIENTIFIC
     ACTIVITIES DIR

DIRECTOR
SAFEGUARD SYSTEM MANAGER
USA BALLISTIC MISSILE DEFENSE
  PROGRAM OFFICE
1330 WILSON BLVD
ARLINGTON, VA 22209
  ATTN REENTRY PHYSICS, J. J. SHEA
  ATTN RES & ENG. MAJ. TEALE,
    DAC-SAE-S

COMMANDER
SAFEGUARD SYSTEM SITE ACTIVATION
  COMMAND
GRAND FORKS
P.O. BOX 631
LANGDON, ND 58249
  ATTN DEP FOR SITE ACTIVATION

COMMANDER
USA BALLISTIC MISSILE DEFENSE
  SYSTEMS COMMAND
P.O. BOX 1500
HUNTSVILLE, AL 35807
  ATTN BMDSC-STP, J. H. DAUGHTRY
  ATTN BMDSC-STE
  ATTN BMDSC-DEM
  ATTN BMDSC-DEM, L. L. DICKERSON
  ATTN BMDSC-DH

COMMANDER
USA SAFEGUARD SYSTEM COMMAND,
  FIELD OFFICE
BELL TELEPHONE LABORATORIES,
  WHIPPANY ROAD
WHIPPANY, NJ 07981
  ATTN SSC-DEF-B, J. TURNER
    SAFCOM PROJ ENGINEER

COMMANDER
USA SAFEGUARD SYSTEM EVALUATION AGENCY
WHITE SANDS MISSILE RANGE, NM 88002
  ATTN SAFSEA-EAB
  ATTN SSEA-RB

COMMANDER
US ARMY MATERIEL COMMAND
REDSTONE ARSENAL, AL 35809
  ATTN AMCPM-LCES, H. HENDRICKSON
  ATTN AMCPM-MD, SAM-D PROJ OFC
  ATTN AMCPM-MDE
  ATTN AMCPM-MDE, MAJ STANLEY

COMMANDER
US ARMY MATERIEL COMMAND
FORT MONMOUTH, NJ 07703
  ATTN AMCPM-TDS, PROJ MGR,
    ARMY TACTICAL DATA SYSTEMS (ARTADS)
  ATTN AMCPM-TDS-TF

COMMANDER
USA COMMUNICATIONS SYSTEMS AGENCY
FORT MONMOUTH, NJ 07703
  ATTN LIBRARY
  ATTN SEYMOUR KREVSKY, DEPT DIR ENGR

COMMANDER
USA STRATEGIC COMMUNICATIONS COMMAND
FORT HUACHUCA, AZ 85613

COMMANDER
ARMY MATERIALS & MECHANICS RESEARCH CENTER
WATERTOWN, MA 02172
  ATTN AMXMR-XH, JOHN DIGNAM

COMMANDER
USA FOREIGN SCIENCE & TECHNOLOGY CENTER
FEDERAL OFFICE BUILDING
220 7TH SREEET NE
CHARLOTTESVILLE, VA 22901
  ATTN AMXST-TDI, DR. P. A. CROWLEY
  ATTN AMXST-ISI, D. McCALLUM
  ATTN LIBRARY SERVICES BRANCH

COMMANDER
USA SATELLITE COMMUNICATIONS AGENCY
FORT MONMOUTH, NJ 07703
  ATTN AMCPM-SC-6, MR. PERLE

COMMANDER
USAMC ABERDEEN RESEARCH & DEV CENTER
ABERDEEN PROVING GROUND, MD 21005
  ATTN AMXRD-BVL, J. H. McNEILLY
  ATTN AMXRD-BVL, J. W. KINCH

COMMANDER
USA ELECTRONICS COMMAND
FORT MONMOUTH, NJ 07703
  ATTN AMSEL TL NN, DR. E. BOTH
  ATTN AMSEL-TL-ND, E. T. HUNTER
  ATTN AMSEL-TL-D, H. K. ZIEGLER
  ATTN AMSEL-GG-TD, SARAH OMANSON
  ATTN AMSEL-NL-D
  ATTN AMSEL-TL-D
  ATTN AMSEL-TL-ME, M. POMERANTZ
  ATTN AMSEL-GG-M, G. K. GAULE
  ATTN AMSEL-TL-NR, DR. H. A. BOMKE
  ATTN AMSEL-WL-D
  ATTN AMSEL-CT-HDK, COHEN.
  ATTN AMSEL-TL-N, DR. S. KRONENBERG
  ATTN AMSEL-TL-NS, R. FREIBERG

COMMANDER
USA ELECTRONICS COMMAND
FORT BELVOIR, VA 22060
  ATTN AMSEL-NV, CPT PARKER

COMMANDER
USA MISSILE COMMAND
REDSTONE ARSENAL, AL 35809
  ATTN AMSMI-RBLD, CHIEF DOC SECTION
  ATTN AMCPM-HA, HAWK PROJ OFC
  ATTN AMSMI-RGG, J. HOLEMAN
  ATTN AMSMI-RR, MR. LIVELY
  ATTN AMCPM-PE-EA, S. D. COZBY

COMMANDER
USA MOBILITY EQUIPMENT R & D CENTER
FORT BELVOIR, VA 22060
  ATTN SMEFB-MW, J. W. BOND
  ATTN SMEFB-BA, R. K. YOUNG
  ATTN SMEFB-RN, D. B. DINGER
  ATTN T. W. O'CONNOR, JR.
  ATTN SMEFB-HHD, F. P. GOOD
  ATTN SMEFB-XN, W. J. HAAS
  ATTN SMEFB-ES, R. S. BRANTLY, JR.

COMMANDER
USA MUNITIONS COMMAND
DOVER, NJ 07801
  ATTN AMSMU-RE-CN, SYS DEV DIV,
    CHEMICAL & NUCLEAR, MR. WAXLER

COMMANDER
PICATINNY ARSENAL
DOVER, NJ 07801
  ATTN SARPA-FR-E, L. AVRAMI
  ATTN SARPA-FR-S
  ATTN SARPA-TS-T-S, TECHNICAL LIBRARY
  ATTN SARPA-ND, P. ZIRKIND
  ATTN SARPA-ND-NE
  ATTN SARPA-ND-C-S, DR. AMINO NORDIO
  ATTN SARPA-ND-DC2
  ATTN SARPA-ND-DB, E. J. ARBER
  ATTN SARPA-ND-DA3
  ATTN SARPA-NDB 300, BLDG 95,
    ARTHUR NICHOLS
  ATTN SARPA-RT-S, FOR JAWTIP
  ATTN SARPA-ND-W
  ATTN SARPA-QA-N, P. G. OLIVIERI
  ATTN SARPA-TS-I-E, A. GRINOCH
  ATTN HYMAN POSTERNAK

COMMANDER
USA TEST & EVALUATION COMMAND
ABERDEEN PROVING GROUND, MD 21005
  ATTN AMSTE-EL, R. I. KOLCHIN
  ATTN AMSTE-NB, R. R. GALASSO

COMMANDER
USA ABERDEEN PROVING GROUND
ABERDEEN PROVING GROUND, MD 21005
  ATTN STEAP-TL, USAARDC BR(BRL) BLD 330

PRESIDENT
USA AIRBORNE COMMUNICATIONS & ELECT BD
FORT BRAGG, NC 28307
  ATTN STEB-MA-A

COMMANDER
USA ELECTRONCIS PROVING GROUND
FORT HUACHUCA, AZ 85613
  ATTN STEEP-MT-G
  ATTN STEEP-MT-M, MR. MCINTOSH

COMMANDER
WHITE SANDS MISSILE RANGE, NM 88002
  ATTN STEWS-TE-N, M. P. SQUIRES

COMMANDER
USACDC CONCEPTS & FORCE DESIGN GROUP
HOFFMAN BUILDING
2461 EISENHOWER AVENUE
ALEXANDRIA, VA 22314
  ATTN CDCCONF-MTCS

COMMANDER
USA CDC NUCLEAR AGENCY
FORT BLISS, TX 79916
  ATTN CDCNA-E

COMMANDER
USACDC ARMOR AGENCY
FORT KNOX, KT 40121
  ATTN DOCUMENT CONTROL

COMMANDER
USA CDC COMMUNICATIONS-ELECTRONICS
 AGENCY
FORT MONMOUTH, NJ 07703
  ATTN CHIEF, M/E DIV

COMMANDER
USA COMPUTER SYSTEMS COMMAND
FORT BELVOIR, VA 22060
  ATTN CSCS-EME-E, F. T. PARKER
  ATTN CSCS-EME-C

CHIEF OF ENGINEERS
DEPARTMENT OF THE ARMY
WASHINGTON, DC 20314
  ATTN DAEN-MCE-D, MR. McCAULEY

DIVISION ENGINEER
USA ENGINEER DIVISION,
 MISSOURI RIVER
P.O. BOX 103 DOWNTOWN STATION
OMAHA, NE 68101
  ATTN MRDED-MC, F. L. HAZLETT,
      SPEC PROJ COORDINATOR

COMMANDER
ARMY NUCLEAR & CHEMICAL SURETY GROUP
FORT BELVOIR, VA 22060
  ATTN FDSG-ND, BLDG 2073, NORTH AREA

COMMANDER
US ARMY SECURITY AGENCY
ARLINGTON HALL STATION
ARLINGTON, VA 22212
  ATTN IARD-EL
  ATTN SPECIAL PROJECTS ELEMENT

COMMANDER
USA STRATEGIC COMMUNICATIONS COMMAND
FORT HUACHUCA, AZ 85613
  ATTN SCCX-SSA-OD
  ATTN SCCX-SSA, COL H. J. STIRLING
  ATTN ACC-FD-C/EMP

COMMANDER
USA COMMAND & GENERAL STAFF COLLEGE
FORT LEAVENWORTH, KN 66027
  ATTN ATSCS-SE-L

COMMANDER
USA FIELD ARTILLERY SCHOOL
FORT SILL, OK 73503
  ATTN ATSFA-CA-NW

CHIEF OF NAVAL OPERATIONS
NAVY DEPARTMENT
WASHINGTON, DC 20350
  ATTN NOP-985F2, CDR. S. I. STOCKING
  ATTN NOP-932
  ATTN NOP-03EG

CHIEF OF NAVAL RESEARCH
DEPARTMENT OF THE NAVY
ARLINGTON, VA 22217
  ATTN ONR-427
  ATTN ONR-418, G. R. JOINER

COMMANDER
NAVAL AIR SYSTEMS COMMAND, HQ
1411 JEFFERSON DAVIS HIGHWAY
ARLINGTON, VA 20360
  ATTN NAIR-350-F, LCDR HUGO HARDT

COMMANDING OFFICER
NAVAL AMMUNITION DEPOT
CRANE, IN 47522
  ATTN CODE 7024, JAMES L. RAMSEY

COMMANDING OFFICER
NAVAL CIVIL ENGINEERING LABORATORY
PORT HUENEME, CA 93043
  ATTN CODE L31

COMMANDER
NAVAL COMMUNICATIONS COMMAND, HQ
4401 MASSACHUSETTS AVE, NW
WASHINGTON, DC 20390
  ATTN N-7, LCDR HALL

COMMANDER
NAVAL ELECTRONICS SYSTEMS
 COMMAND, HQ
2511 JEFFERSON DAVIS HIGHWAY
ARLINGTON, VA 20360
  ATTN NELEX-05123
  ATTN NELEX-5124, BERT FOX
  ATTN NELEX-0518

COMMANDER
NAVAL ELECTRONICS LABORATORY CENTER
SAN DIEGO, CA 92152
  ATTN TECHNICAL LIBRARY
  ATTN CODE 3100, E. E. McCOWN
  ATTN CODE 3200, J. F. WONG

COMMANDER
NAVAL INTELLIGENCE SUPPORT CENTER
4301 SUITLAND ROAD
WASHINGTON, DC 20390
  ATTN DR. P. ALEXANDER
  ATTN NISC-41

COMMANDER
NAVAL SURFACE WEAPONS CENTER
WHITE OAK, MD 20910
  ATTN CODE 121, NUCLEAR PROGRAM OFC
  ATTN CODE 244, EXPLOSIONS EFFECTIVENESS
      DIVISION
  ATTN CODE 730, LIBRARY DIVISION
      (6 COPIES)
  ATTN CODE 431, NORMAN TASLITT
  ATTN M. C. PETREE
  ATTN R. A. SMITH
  ATTN E. R. RATHBUN

COMMANDER
NAVAL SEA SYS COM
2521 JEFFERSON DAVIS HIGHWAY
ARLINGTON, VA 20360
  ATTN NSEA-0523, R. LANE

SUPERINTENDENT
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93940
  ATTN CODE 2124, LIBRARY

DIRECTOR
NAVAL RESEARCH LABORATORY
WASHINGTON, DC 20390
  ATTN CODE 4004, E. L. BRANCATO
  ATTN CODE 6465, DR. RICHARD L. STATLER
  ATTN CODE 7001, J. D. BROWN

COMMANDING OFFICER
NAVAL SCIENTIFIC AND TECHNICAL
 INTELLIGENCE CENTER
4301 SUITLAND ROAD, BLDG 5
WASHINGTON, DC 20390
  ATTN DOCUMENT CONTROL

COMMANDER
NAVAL SHIP ENGINEERING CENTER
3700 EAST-WEST HIGHWAY
PRINCE GEORGES PLAZA
HYATTSVILLE, MD 20782
  ATTN NSEC-6174D2, EDWARD F. DUFFY
  ATTN NSEC-6015C, EDWARD BERKOWITZ

COMMANDER
NAVAL SEA SYS COM
2531 JEFFERSON DAVIS HIGHWAY
ARLINGTON, VA 20360
  ATTN NSHP-03541, WILLIAM S. BROWN

COMMANDER
NAVAL UNDERSEA CENTER
SAN DIEGO, CA 92132
  ATTN CODE 608, C. F. RANSTEDT

COMMANDER
NAVAL WEAPONS CENTER
CHINA LAKE, CA 93555
  ATTN CODE 753, LIBRARY DIVISION

NAVAL WEAPONS ENGINEERING SUPPT ACTVY
WASHINGTON NAVY YARD
WASHINGTON, DC 20390
  ATTN E, S, A, 70

COMMANDING OFFICER
NAVAL WEAPONS EVALUATION FACILITY
KIRTLAND AFB, NM 87117
  ATTN CODE WE, MR. STANLEY
  ATTN L. OLIVER

COMMANDER
NAVAL SURFACE WEAPONS CENTER
DAHLGREN, VA 22448
  ATTN WILLIAM H. HOLT

COMMANDING OFFICER
NAVAL WEAPONS STATION
CONCORD, CA 94520
  ATTN QUAL EVAL LAB, CODE 33120
      DR. ROBERT WAGNER

COMMANDER
NAVY ASTRONAUTICS GROUP
POINT MUGU, CA 93042
  ATTN WILLIAM GLEESON

COMMANDING OFFICER
NAVY SPACE SYSTEM ACTIVITY
BOX 92960, WORLDWAYS POSTAL CENTER
LOS ANGELES, CA 90009
  ATTN DR. E. E. MUEHLNER

COMMANDER
NUCLEAR WEAPONS TRAINING CTR, PACIFIC
NAVAL AIR STATION, NORTH ISLAND
SAN DIEGO, CA 92135
  ATTN CODE 52

COMMANDER
NUCLEAR WPNS TRAINING GROUND, ATLANTIC
NORFOLK, VA 23511
  ATTN DOCUMENT CONTROL

DIRECTOR
STATEGIC SYSTEMS PROJECTS OFFICE
NAVY DEPARTMENT
WASHINGTON, DC 20390
  ATTN NSP-431, TECHNICAL LIBRARY
  ATTN NSP-230, DAVID GOLD
  ATTN NSP-2342, R. L. CODLMAN
  ATTN NSP-2701,JOHN W. PITSENBERGER
  ATTN NSP-273, PHIL SPECTOR
  ATTN NSP-2701, CDR L. STOESSL

COMMANDER-IN-CHIEF
US ATLANTIC FLEET
NORFOLK, VA 23511
  ATTN DOCUMENT CONTROL

COMMANDER-IN-CHIEF
US PACIFIC FLEET
FPO SAN FRANCISCO, CA 96610
  ATTN DOCUMENT CONTROL (303)

COMMANDER
NAVAL TELECOMMUNICATIONS COMMAND
4401 MASSACHUSETTS AVE. NW
WASHINGTON, DC 20390
  ATTN DEPUTY DIRECTOR (SYSTEMS)

CHIEF OF STAFF
US AIR FORCE, HQ
WASHINGTON, DC 20330
  ATTN RDQPN (S/V & NUCLEAR
       PROGRAMS DIV.)
  ATTN PRCCS, LTC WOODRUFF
  ATTN RDPS, MR. PORTER
  ATTN IGSPB, ED CALVERT

ASSISTANT DEPUTY CHIEF OF STAFF
RESEARCH & DEVELOPMENT
HEADQUARTERS, U.S. AIR FORCE
WASHINGTON, DC 20330
  ATTN RDQ/SN

COMMANDER
AEROSPACE DEFENSE COMMAND
ENT AFB, CO 80912
  ATTN XPDW, ADVANCED PLANNING DIV.
  ATTN DEEOS, J. C. BRANNAN

COMMANDER
AIR UNIVERSITY
MAXWELL AFB, AL 36112
  ATTN AUL/LSE-70-250

COMMANDER
HQ AIR FORCE SYSTEMS COMMAND
ANDREWS AFB
WASHINGTON, DC 20331
  ATTN DLSP (GENERAL PHYSICS)

COMMANDER
AF AERO PROPULSION LABORATORY, AFSC
WRIGHT-PATTERSON AFB, OH 45433
  ATTN P. E. STOVER

COMMANDER
AEROSPACE RESEARCH LABORATORIES, AFSC
WRIGHT-PATTERSON AFB, OH 45433
  ATTN CA
  ATTN LS, DONALD C. REYNOLDS

COMMAND
AF AVIONICS LABORATORY, AFSC
WRIGHT-PATTERSON, OH 45433
  ATTN AAA, AVIONICS SYNTHESIS & ANAL BR
  ATTN NVS, EUGENE C. MAUPIN
  ATTN NVS, R. CONKLIN
  ATTN TEA, DR. HANS HENNECKE

COMMANDER
AF FLIGHT DYNAMICS LAB
WRIGHT-PATTERSON AFB, OH 45433
  ATTN RUDY BEAVIN

COMMANDER
AF WEAPONS LABORATORY, AFSC
KIRTLAND AFB, NM 87117
  ATTN ELT, MAJ WALKER
  ATTN SAA
  ATTN SAB
  ATTN SAY
  ATTN EL, CAPT CARL DAVIS
  ATTN EL, J. DARRAH
  ATTN ELE/EMP BRANCH
  ATTN SUL, TECHNICAL LIBRARY
  ATTN ELE, DR. CARL BAUM

COMMANDER
ROME AIR DEVELOPMENT CENTER, AFSC
GRIFFISS AFB, NY 13440
  ATTN RCRP, J. S. SMITH
  ATTN TSGC
  ATTN RCRM, CAPT R. BELLEM

COMMANDER
ARMAMENT DEVELOPMENT AND TEST CENTER
EGLIN AIR FORCE BASE, FL 32542
  ATTN ADTC(DLO6L), TECH LIBRARY

COMMANDER
AERONAUTICAL SYSTEMS DIVISION, AFSC
WRIGHT-PATTERSON AFB, OH 45433
  ATTN ASD/YH-EX, CAPT BRANHAM
  ATTN ASD/ENVCB, ROBERT L. FISH
  ATTN ASD/ENVED

COMMANDER
ELECTRONICS SYSTEMS DIVISION, AFSC
L. G. HANSCOM FIELD
BEDFORD, MA 01730
  ATTN MCAE, LTC DAVID SPARKS
  ATTN XRE, SURVIVABILITY
  ATTN LCD
  ATTN YWES
  ATTN DCKE
  ATTN XRP, MAJ GINGRICH
  ATTN IN
  ATTN DCD
  ATTN MCL

COMMANDER
FOREIGN TECHNOLOGY DIVISION, AFSC
WRIGHT-PATTERSON AFB, OH 45433
  ATTN PDTN, BALLARD
  ATTN FTD/PDJC

COMMANDER
HQ SPACE AND MISSILE SYSTEMS ORGANIZATION
P.O. 96960 WORLDWAYS POSTAL CENTER
LOS ANGELES, CA 90009
  ATTN RSS SYSTEM ENGINEERING
  ATTN SKD
  ATTN SKE, DIR OF ENGR GP 1
  ATTN SKT
  ATTN IND, I. J. JUDY
  ATTN XRT, STRATEGIC SYSTEMS DIV
  ATTN SYJ, AEROSPACE DEFENSE PROG OFC
  ATTN SZH, CAPT MARION F. SCHNEIDER
  ATTN SZJ, CAPT E. T. DEJONCKHEERE, JR.
  ATTN CCD, CAPT A. MENDEKE, JR.
  ATTN DYS, MAJ HEILMAN
  ATTN RSP, SYSTEM DEFN & ASSESSMENT,
       LTC GILBERT
  ATTN DYJ, CAPT RASMUSSEN
  ATTN RNF, MAJ COX

SPACE & MISSILE SYSTEMS ORGANIZATION
NORTON AFB, CA 92490
  ATTN SYGN, CAPT STROBEL
  ATTN RSTA, E. A. MERRITT

AF INSTITUTE OF TECHNOLOGY, AU
WRIGHT-PATTERSON AFB, OH 45433
  ATTN AFIT(ENP), DR. CHARLES J. BRIDGMAN

HEADQUARTERS
AIR FORCE TECHNICAL APPLICATIONS CENTER
PATRICK AFB, FL 32925
  ATTN TD-5B
  ATTN TD-3

COMMANDER
STRATEGIC AIR COMMAND
OFFUTT AFB, NB 68113
  ATTN NRI, STINFO LIBRARY
  ATTN NRW

US ATOMIC ENERGY COMMISION
WASHINGTON, DC 20545
  ATTN DIVISION OF HQ SERVICES
       LIBRARY BRANCH, RALPH SHULL, DMA
  ATTN DMA, DOCUMENT CONTROL FOR
       R&D BRANCH

US ATOMIC ENERGY COMMISSION
ALBUQUERQUE OPERATIONS OFFICE
P.O. BOX 5400
ALBUQUERQUE, NM 87115
  ATTN DOCUMENT CONTROL

ADMINISTRATOR
DEFENSE ELECTRIC POWER ADMINISTRATION
DEPT OF INTERIOR
WASHINGTON, DC 20240
  ATTN IRVING, I. RAINES, ROOM 5600

DEPARTMENT OF COMMERCE
NATIONAL BUREAU OF STANDARDS
WASHINGTON, DC 20234
  ATTN ELECTRON DEV SECT, J. C. FRENCH

NATIONAL ACADEMY OF SCIENCES
2101 CONSTITUTION AVE., NW
WASHINGTON, DC 20418
  ATTN DR. R. S. SHANE
       NATIONAL MATERIALS ADVISORY BOARD

NASA HEADQUARTERS
WASHINGTON, DC 20546
  ATTN CODE REE, GUIDANCE, CONTROL AND
       INFORMATION SYSTEMS

ARMS CONTROL AND DISARMAMENT AGENCY
REFERENCE INFORMATION CENTER
DEPARTMENT OF STATE
2201 C STREET, NW
WASHINGTON, DC 20451
  ATTN CRSC/L, AUDREY E. EDMONDS

FEDERAL AVIATION ADMINISTRATION
DEPARTMENT OF TRANSPORTATION
800 INDEPENDENCE AVENUE, SW
WASHINGTON, DC 20590
  ATTN F. S. SAKATE, RD 650

UNIVERSITY OF CALIFORNIA
LAWRENCE LIVERMORE LABORATORY
TECHNICAL INFORMATION DIVISION
P.O. BOX 808
LIVERMORE, CA 94551
  ATTN L-48, DR. LOUIS F. WOUTERS
  ATTN L-3, TECHNICAL INFO DEPT
  ATTN L-31, WILLIAM J. HOGAN

DISTRIBUTION (Cont'd)

UNIVERSITY OF CALIFORNIA
LAWRENCE LIVERMORE LABORATORY (Cont'd)
  ATTN L-24, DR. DAVID OAKLEY
  ATTN L-24, HANS KRUGER
  ATTN L-156, L. L. CLELAND
  ATTN L-71, DR. W. GRAYSON
  ATTN L-156, E. K. MILLER

UNIVERSITY OF CALIFORNIA
LAWRENCE RADIATION LABORATORY
LIBRARY, BUILDING 50, RM 134
BERKELEY, CA 94720
  ATTN PROF. KENNETH M. WATSON

UNIVERSITY OF CALIFORNIA
  ATTN DOCUMENT CONTROL
LOS ALAMOS SCIENTIFIC LABORATORY
P.O. BOX 1663
LOS ALAMOS, NM 87544
  ATTN GMX-7, TERRY R. GIBBS
  ATTN DR. JOHN S. MALIK
  ATTN J-DOT FOR DR. RALPH PARTRIDGE
  ATTN R. F. TASCHEK
  ATTN J. ARTHUR FREED

UNIVERSITY OF DENVER
COLORADO SEMINARY
DENVER RESEARCH INSTITUTE
  ATTN SECURITY OFFICER
P.O. BOX 10127
DENVER, CO 80210
  ATTN FRED P. VENDITTI
  ATTN R. W. BUCHANAN

GEORGIA INSTITUTE OF TECHNOLOGY
OFFICE OF RESEARCH ADMINISTRATION
ATLANTA, GA 30332
  ATTN RES & SFC COORD FOR H. DENNY

IIT RESEARCH INSTITUTE
10 WEST 35TH STREET
CHICAGO, IL 60616
  ATTN J. E. BRIDGES, ENGR ADVISOR
  ATTN J. J. KRSTANSKY, ASST DIR OF RSCH
  ATTN I. N. MINDEL

IIT RESEARCH INSTITUTE
ELECTROMAGNETIC COMPATABILITY
  ANALYSIS CENTER
NORTH SEVERN-ECAC BLDG
ANNAPOLIS, MD 21402
  ATTN ACOAT

MIT LINCOLN LABORATORY
P.O. BOX 73, 244 WOOD STREET
LEXINGTON, MA 02173
  ATTN ALAN G. STANLEY

C. S. DRAPER LABORATORY DIVISION
  OF MASSACHUSETTS
  INSTITUTE OF TECHNOLOGY
68 ALBANY STREET
CAMBRIDGE, MA 02139
  ATTN KENNETH FERTIG, MS-87

TEXAS TECH UNIVERSITY
P.O. BOX 5404 NORTH COLLEGE STA
LUBBOCK, TX 79409
  ATTN TRAVIS L. SIMPSON

AMERICAN TELEPHONE & TELEGRAPH
ADMINISTRATION OFFICE
2055 L ST. NW
WASHINGTON, DC 20036
  ATTN M.R. GRAY FOR
    W. L. EDWARDS

AEROJET ELECTRO-SYSTEMS CO. DIV
AEROJET-GENERAL CORPORATION
P.O. BOX 296
AZUSA CA 91702
  ATTN T. D. HANSCOME, 6181/170
  ATTN R. Y. KAKUDA, B-194/D-6121

AEROJET ENERGY CONVERSION COMPANY
AEROJET LIQUID ROCKET COMPANY
P.O. BOX 13222
SACRAMENTO, CA 95813
  ATTN DEPT. 8130

AEROJET ENERGY CONVERSION COMPANY
AEROJET NUCLEAR SYSTEMS COMPANY
P.O. BOX 13070
SACRAMENTO, CA 95813
  ATTN TECH LIB, DEPT N4264,
                    BLDG 2019A1

AEROSPACE CORPORATION
E. EL SEGUNDO BLVD
EL SEGUNDO, CA 90245
  ATTN FRANCIS HAI
  ATTN DIR, OFF OF TECH SURVIV,
    V. JOSEPHSON
  ATTN DIR, SAT SYS DIV, GP II,
    V. WALL
  ATTN DIR, SAT SYS DIV, GP IV,
    F. KELLER
  ATTN LIBRARY ACQUISITION GROUP
  ATTN WPNS EFF DEPT,
    DR. J. REINHEIMER
  ATTN NUCLEAR & ENVIRONMENTAL
    STAFF, N. D. STOCKWELL
  ATTN DR. JERRY COMISAR
  ATTN DIR, HARDENED REENTRY SYSTEMS,
    R. MORTENSEN
  ATTN J. BENVENISTE

AMERICAN NUCLEONICS CORP.
  ATTN SECURITY OFFICER
6036 VARIEL AV.
WOODLAND HILLS, CA 91364
  ATTN GLENN L. BROWN, DIR OF RES
  ATTN DR. R. N. GHOSE

ARINC RESEARCH CORPORATION
WESTERN DIVISION
1222 E. NORMANDY PLACE
P.O. BOX 1375
SANTA ANA, CA 92702
  ATTN DEPT MGR ENG,
    J. M. ALDERMAN

ART RESEARCH CORPORATION
1100 GLENDON AVENUE
LOS ANGELES, CA 90024
  ATTN ARTHUR SANDERS
  ATTN T. JORDAN

ASTRONAUTICS CORP OF AMERICA
907 SOUTH FIRST ST
MILWAUKEE, WI 53204
  ATTN T. KERN

AVCO SYSTEMS DIVISION
201 LOWELL STREET
WILMINGTON, MA 01887
  ATTN RESEARCH LIBRARY A220,
    RM 2201

AVCO CORPORATION
ELECTRONICS DIVISION
2630 GLENDALE-MILFORD ROAD
CINCINNATI, OH 45241
  ATTN RON GOLDFARB

AVCO-EVERETT RESEARCH LABORATORY
2385 REVERE BEACH PARKWAY
EVERETT, MA 02149
  ATTN LORAIN NAZZARD

BATTELLE MEMORIAL INSTITUTE
505 KING AVENUE
COLUMBUS, OH 43201
  ATTN R. K. THATCHER
  ATTN STOIAC

BEECH AIRCRAFT CORPORATION
9709 EAST CENTRAL AVENUE
WICHITA, KS 67201
  ATTN EDWARD L. RADELL

BELL AEROSPACE COMPANY
DIVISION OF TEXTRON, INC.
P.O. BOX 1
BUFFALO, NY 14240
  ATTN MS F-11, MARTIN A. HENRY
  ATTN CARL B. SCHOCH, WPNS EFFECTS GP

BELL TELEPHONE LABORATORIES, INC.
MOUNTAIN AVENUE
MURRAY HILL, NJ 07974
  ATTN H. JARRELL, RM WH-2F-153
  ATTN I. G. DURAND
  ATTN R. D. TAFT, RM 2B-181
  ATTN A. H. CARTER, WH-4522
  ATTN H.J. BETZEL
  ATTN FRANK P. ZUPA

BELL TELEPHONE LABORATORIES, INC.
INTERSTATE 85 AT MT. HOPE CHURCH ROAD
P.O. BOX 21447
GREENSBORO, NC 27420
  ATTN CHARLES E. BOYLE
  ATTN JAMES F. SWEENEY

BENDIX CORPORATION, THE
AEROSPACE SYSTEMS DIVISION
3300 PLYMOUTH ROAD
ANN ARBOR, MI 48107
  ATTN MR. RONALD H. PIZAREK

BENDIX CORPORATION, THE
COMMUNICATION DIVISION
JOPPA ROAD, TOWSON
BALTIMORE, MD 21204
  ATTN DOCUMENT CONTROL

BENDIX CORPORATION
RESEARCH LABORATORIES DIVISION
BENDIX CENTER
SOUTHFIELD, MI 48075
  ATTN MANAGER, PROGRAM DEVELOPMENT
    MR. DONALD J. NIEHAUS

BENDIX CORPORATION, THE
NAVIGATION AND CONTROL DIVISION
TETERBORO, NJ 07608
  ATTN E. E. LADEMANN
  ATTN T. LAVIN, DEPT 7111
  ATTN CHIEF LIBRARIAN, LYDIA FARRELL

BOEING COMPANY, THE
P.O. BOX 3999
SEATTLE, WA 98124
  ATTN D. L. DYE, 2-6005, 45-21

159

DISTRIBUTION (Cont'd)

BOEING COMPANY, THE
P.O. BOX 3707
SEATTLE, WA 98124
  ATTN R. S. CALDWELL, MS 2R-00
  ATTN H. W. WICKLEIN, MS 1F-51
  ATTN A. R. LOWREY, MS 2R-00
  ATTN AEROSPACE LIBRARY
  ATTN B. L. CARLSON, MS 4240
  ATTN MR. E. NOWAK, MS 47/14

BOOZ-ALLEN APPLIED RESEARCH, INC.
106 APPLE STREET
NEW SHREWSBURY, NJ 07724
  ATTN FREDERICK NEWTON

BRADDOCK, DUNN & McDONALD, INC.
P.O. BOX 8885 STATION C
ALBUQUERQUE, NM 87108
  ATTN ROBERT B. BUCHANAN

BRADDOCK, DUNN & McDONALD, INC.
8027 LEESBURG PIKE
MCLEAN, VA 22101
  ATTN J. V. BRADDOCK
  ATTN J. J. KALINOWSKI

BRADDOCK, DUNN & McDONALD, INC.
1920 ALINE AVE
VIENNA, VA 22180
  ATTN DR. J. BRADDOCK

BROWN ENGINEERING COMPANY, INC.
RESEARCH PARK
HUNTSVILLE, AL 35807
  ATTN D. LAMBERT, MS 126

CALSPAN CORPORATION
P.O. BOX 235
BUFFALO, NY 14221
  ATTN R. H. DICKHAUT, BLDG 10, RM 341

CHRYSLER CORPORATION
DEFENSE DIVISION
P.O. BOX 757
DETROIT, MI 48231
  ATTN R. F. GENTILE, CIMS 435-01-21

COLLINS RADIO COMPANY
5225 C AVENUE, N.E.
CEDAR RAPIDS, IO 52406
  ATTN E. E. ELLISON , LIBRARIAN

COMPUTER SCIENCES CORPORATION
P.O. BOX 530
FALLS CHURCH, VA 22046
  ATTN JOHN D. ILLGEN

CUTLER HAMMER, INC.
AIL DIVISION, COMAC ROAD
DEER PARK, NY 11729
  ATTN CENTRAL TECHNICAL FILS
    ANNE ANTHONY
  ATTN RICHARD J. MOHR

DIKEWOOD CORPORATION, THE
1009 BRADBURY DRIVE, S.E.
UNIVERSITY RESEARCH PARK
ALBUQUERQUE, NM 87106
  ATTN LLOYD WAYNE DAVIS

E-SYSTEMS INC.
GREENVILLE DIVISION
MAJOR FIELD
P.O. BOX 1056
GREENVILLE, TX 75401
  ATTN LIBRARY

EFFECTS TECHNOLOGY, INC.
5383 HOLISTER AVENUE
SANTA BARBARA, CA 93105
  ATTN EDWARD JOHN STEELE

EG&G, INC..
SAN RAMON OPERATIONS
P.O. BOX 204
SAN RAMON, CA 94583
  ATTN BURNELL G. WEST

EG&G, INC.
P.O. BOX 4339
ALBUQUERQUE, NM 87106
  ATTN WEYLAND D. GEORGE
  ATTN HILDA HOFFMAN

EMERSON ELECTRIC COMPANY
8100 FLORISSANT
ST. LOUIS, MO 63136
  ATTN DOCUMENT CONTROL

ENERGY CONVERSION DEVICES, INC.
1675 WEST MAPLE ROAD
TROY, MO 48084
  ATTN LIONEL ROBBINS

FAIRCHILD INDUSTRIES
SHERMAN FAILCHILD TECHNOLOGY CENTER
20301 CENTURY BOULEVARD
GERMANTOWN, MD 20767
  ATTN LEONARD J. SCHREIBER

FAIRCHILD CAMERA & INSTRUMENT CORP
464 ELLIS STREET
MOUNTAIN VIEW, CA 94040
  ATTN SECURITY DEPT FOR 30-204
    DAVID K. MYERS

GARRETT CORPORATION, THE
9851 SEPULVEDA BLVD
LOS ANGELES, CA 90009
  ATTN ROBER WEIR, DEPT 93-9

GENERAL DYNAMICS CORPORATION
CONVAIR AEROSPACE DIVISION
SAN DIEGO OPERATIONS
P.O. BOX 1950
SAN DIEGO, CA 92112
  ATTN V. J. SWEENEY, INTERDIV
    RESEARCH

GENERAL ELECTRIC COMPANY
APOLLO & GROUND SYSTEMS, HOUSTON
1830 NASA BOULEVARD
P.O. BOX 58408
HOUSTON, TX 77058
  ATTN H. E. SHARP

GENERAL ELECTRIC COMPANY
P.O. BOX 1122
SYRACUSE, NY 13201
  ATTN HMES, BLDG 1, RM 4,
    J. R. GREENBAUM
  ATTN CSP 6-7, L. H. DEE

GENERAL ELECTRIC COMPANY
AEROSPACE ELECTRONICS DEPARTMENT
FRENCH ROAD
UTICA, NY 13502
  ATTN W. J. PATTERSON, DROP 233
  ATTN MAIL STA 624, FRED NICOTERA

GENERAL ELECTRIC COMPANY
P.O. BOX 5000
BINGHAMTON, NY 13902
  ATTN B. H. SHOWALTER, MD 160

GENERAL ELECTRIC COMPANY
ORDNANCE SYSTEMS
100 PLASTICS AVENUE
PITTSFIELD, MA 01201
  ATTN DAVID CORMAN, MN 2276

GENERAL ELECTRIC COMPANY
100 WOODLAWN AVE
PITTSFIELD, MA 01201
  ATTN FRANK FISHER, BLDG9-209

GENERAL ELECTRIC COMPANY
RE-ENTRY & ENVIRONMENTAL SYSTEMS DIVISION
P.O. BOX 7722
PHILADELPHIA, PA 19101
  ATTN ROBERT V. BENEDICT

GENERAL ELECTRIC COMPANY
SPACE DIVISION
VALLEY FORGE SPACE CENTER
P.O. BOX 8555
PHILADELPHIA, PA 19101
  ATTN JOSEPH C. PEDEN, CCF 8301
  ATTN RADIATION EFFECTS LAB, J. L. ANDREWS
  ATTN LIBRARIAN, L. I. CHASEN
  ATTN DANIEL EDELMAN
  ATTN J. P. SPRATT, RM 9421

GENERAL ELECTRIC COMPANY
TEMPO-CENTER FOR ADVANCED STUDIES
816 STATE STREET
SANTA BARBARA, CA 93102
  ATTN DASIAC

GENERAL MOTORS CORPORATION
DELCO ELECTRONICS DIVISION
7929 SOUTH HOWELL AVENUE
OAK CREEK, WI 53201
  ATTN TECHNICAL LIBRARY 2A07
    E. T. KRUEGER

GENERAL RESEARCH CORPORATION
P.O. BOX 3587
SANTA BARBARA, CA 93105
  ATTN TECH INFO OFFICE FOR R. D. HILL

GENERAL RESEARCH CORPORATION
1501 WILSON BLVD
ARLINGTON, VA 22209
  ATTN DR. WILLIAM JOHNSON

GOODYEAR AEROSPACE CORPORATION
ARIZONA DIVISION
LITCHFIELD PARK, AZ 85340
  ATTN B. MANNING

GRUMMAN AEROSPACE CORPORATION
SOUTH OYSTER BAY ROAD
BETHPAGE, NY 11714
  ATTN J. ROGERS, PLANT 35, DEPT 533

GTE SYLVANIA, INC.
77 A STREET
NEEDHAM, MA 02194
  ATTN LIB, C. THORNHILL
  ATTN ELECT SYS GP S/V ENG DEPT,
    J. A. WALDRON
  ATTN J. H. TERRELL
  ATTN ELECT SYST DIV, L. L. BLAISDELL

160

GTE SYLVANIA, INC.
COMMUNICATIONS SYSTEMS DIVISION
189 B STREET
NEEDHAM, MA 02194
  ATTN S.V ENG DEPT, J. A. WALDRON
  ATTN ASM DEPT, E. P. MOTCHOK
  ATTN ASM DEPT, DR. J. H. TERRELL
  ATTN ASM DEPT, S. A. FIERSTON

HAZELTINE CORPORATION
PULASKI ROAD
GREEN LAWN, NY 11740
  ATTN J. B. COLOMBO

HERCULES INCORPORATED
BACCHUS PLANT
P.O. BOX 98
MAGNA, UT 84044
  ATTN 100K-26-W, R. WOODRUFF

HONEYWELL, INC.
GOVERNMENT & AERO PRODUCTS DIV
2600 RIDGWAY ROAD
MINNEAPOLIS, MN 55413
  ATTN RONALD R. JOHNSON, A1391
  ATTN LIBRARY, V. BARTLETT, R3679

HONEYWELL INCORPORATED
AEROSPACE DIVISION
13350 U.S. HIGHWAY 19
ST. PETERSBURG, FL 33733
  ATTN MR. HARRISON H. NOBLE,
    STAFF ENGINEER, MS 725-5A
  ATTN ADVANCED DEVELOPMENT,
    JAMES D. ALLEN, 724-5
  ATTN MS 725-5, R. C. SCHRADER

HONEYWELL, INC.
RADIATION CENTER
2 FORBES ROAD
LEXINGTON, MA 02173
  ATTN TECHNICAL LIBRARY

HUGHES AIRCRAFT COMPANY
CENTINELA AVENUE AND TEALE STREET
CULVER CITY, CA 90230
  ATTN R&D DIVISION,
    DR. DAN BINDER (M.S. D147)
  ATTN MS6/E110, B. W. CAMPBELL

HUGHES AIRCRAFT COMPANY
GROUND SYSTEMS GROUP
1901 WEST MALVERN AVENUE
FULLERTON, CA 92634
  ATTN TECHNICAL LIBRARY, BLDG 600 MS-C-22

INSTITUTE FOR DEFENSE ANALYSES
400 ARMY-NAVY DRIVE
ARLINGTON, VA 22202
  ATTN CLASSIFIED LIBRARY

INTELCOM/RAD TECH
P.O. BOX 80817
SAN DIEGO, CA 92138
  ATTN DR. V. A. J. VAN LINT
  ATTN R. L. MERTZ
  ATTN JAMES A. NABER
  ATTN LEO D. COTTER
  ATTN TERRY M. FLANIGAN
  ATTN DR. E. P. WENAS
  ATTN W. D. SWIFT
  ATTN DR. T. A. TUMOLILLO

INTERNATIONAL BUSINESS MACHINES CORP.
ROUTE 17C
OWEGO, NY 13827
  ATTN D. C. SULLIVAN, DEPT M40, 102-1
  ATTN FRANK FRANKOVSKY

INTERNATIONAL TELEPHONE AND
  TELEGRAPH CORPORATION
500 RIVER ROAD
NUTLEY, NJ 07110
  ATTN DEFENSE-SPACE GROUP, SMTS,
    FRANK JOHNSON
  ATTN ALEXANDER L. RICHARSON

KAMAN SCIENCES CORPORATION
KAMAN NUCLEAR DIVISION
1700 GARDEN OF THE GODS ROAD
P. O. BOX 7463
COLORADO SPRINGS, CO 80907
  ATTN DR. ALBERT P. BRIDGES
  ATTN DR. FRANK H. SHELTON
  ATTN J. R. HOFFMAN
  ATTN N-GAMMA LAB, DON BRYCE

LOCKHEED MISSILES AND SPACE COMPANY
3251 HANOVER STREET
PALO ALTO, CA 94304
  ATTN DR. CLARENCE F. KOOI,
    DEPT 52-11, BLDG 204
  ATTN DR. LLOYD CHASE
  ATTN DR. M. WALT, DEPT 52-10, BLDG 201
  ATTN DR. S. E. SINGER, DEPT 52-20,
    BLDG 20

LOCKHEED MISSILES AND SPACE COMPANY
P.O. BOX 504
SUNNYVALE, CA 94088
  ATTN H. SCHNEEMAN, 81-62
  ATTN M365, DEPT 81-23, BLDG 154
  ATTN L. F. HEARNE, D/81-14
  ATTN R. N. MUNSON, DEPT 81-01,
    BLDG 154
  ATTN W. KOZUMPLIK, TECHNICAL
    INFORMATION CENTER, BLDG 201
  ATTN G. F. HEATH D/81-14 B/154
  ATTN DEPT 85-85, BLDG 154,
    A. C. FELLER
  ATTN L. J. ROSSI, D/81-62 B/150
  ATTN KEVIN McCARTHY

LTV AERO SPACE CORPORATION
AEROSPACE DIV
P.O. BOX 6267
DALLAS, TX 75222
  ATTN TECHNICAL DATA CENTER

MARTIN MARIETTA CORPORATION
DENVER DIVISION
P.O. BOX 179
DENVER, CO 80201
  ATTN 6617 RESEARCH LIBRARY,
    J. R. McKEE
  ATTN SPECIAL PROJECTS MAIL 0130

MARTIN MARIETTA CORPORATION
AEROSPACE DIVISION
P.O. BOX 5837
ORLANDO, FL 32805
  ATTN W. W. MRAS, NP-413
  ATTN ENG LIBRARY, M. C. GRIFFITH,
    MP-30
McDONNEL DOUGLAS CORPORATION
5301 BOLSA AVENUE
HUNTINGON BEACH, CA 92647
  ATTN N. L. ANDRADE, MS 17
    BBDO ADV ELECT/R&D

McDONNELL DOUGLAS CORPORATION
P.O. BOX 516
ST. LOUIS, MO 63166
  ATTN DR. TOM ENDER, DEPT 313,
    BLDG 33
  ATTN LIBRARY

MISSION RESEARCH CORPORATION
735 STATE ST.
P.O. DRAWER 719
SANTA BARBARA, CA 93101
  ATTN C. L. LONMIRE
  ATTN WILLIAM HART

MISSION RESEARCH CORP
P.O. BOX 8693, STATION C
ALBUQUERQUE, NM 87103
  ATTN DAVID E. MEREWETHER
  ATTN JAMES LONERGAN

MITRE CORPORATION, THE
ROUTE 62 AND MIDDLESEX TURNPIKE
P.O. BOX 208
BEDFORD, MA 01703
  ATTN LIBRARY
  ATTN M.E. FITZGERALD
  ATTN THEODORE JARVIS

MOTOROLA, INC.
GOVERNMENT ELECTRONICS DIVISION
8201 EAST McDOWELL ROAD
SCOTTSDALE, AZ 85257
  ATTN PHILIP L. CLAR
  ATTN TECH INFO CENTER-A, J. KORDALEWSKI

NORTH AMERICAN ROCKWELL CORPORATION
3370 MIRALOMA AVENUE
ANAHEIM, CA 92803
  ATTN MINUTEMAN OFC, CA 107,
    D.C. BAUSCH
  ATTN J. BELL
  ATTN G. MESSENGER
  ATTN G. MORGAN
  ATTN J. S. MATYUCH, FA70
  ATTN N. E. AYRES
  ATTN J. SPETZ

NORTH AMERICAN AVIATION-COLUMBUS
NORTH AMERICAN ROCKWELL CORPORATION
4300 EAST FIFTH AVENUE
COLUMBUS, OH 43216
  ATTN ENGINEERING DATA SERVICES,
    J. ROBERTS

NORTH AMERICAN ROCKWELL CORPORATION
LOS ANGELES DIVISION
5601 WEST IMPERIAL HIGHWAY
LOS ANGELES, CA 90009
  ATTN DONALD J. STEVENS, EMI/EMP&RCS
    AVIONICS DESIGN
  ATTN TIC BAOB

NORTH AMERICAN ROCKWELL CORPORATION
SPACE DIVISION
12214 LAKEWOOD BOULEVARD
DOWNEY, CA 90241
  ATTN TIC DEPT 096-AJ01

NORTHROP CORPORATION
NORTHROP RESEARCH & TECHNOLOGY CENTER
3401 WEST BROADWAY
HAWTHORNE, CA 90250
  ATTN DIR, SOLID STATE ELECTRONICS,
    DR. ORLIE L. CURTIS, JR
  ATTN MR. JAMES P. RAYMOND
  ATTN LIBRARY

NORTHROP CORPORATION
ELECTRONIC DIVISION
2301 WEST 120TH STREET
HAWTHORNE, CA 90250
  ATTN BOYCE T. AHLPORT
  ATTN T6114, V. R. DeMARTINO

PALISADES INSTITUTE FOR RESEARCH
  SERVICES, INC.
201 VARICK STREET
NEW YORK, NY 10014
  ATTN RECORDS SUPERVISOR

PHILCO-FORD CORPORATION
AEROSPACE & DEFENSE SYSTEMS OPNS
AERONUTRONIC DIVISION
FORD AND JAMBOREE ROADS
NEWPORT BEACH, CA 92663
  ATTN DR. L. H. LINDER
  ATTN E. R. PONCELET, JR.
  ATTN K. C. ATTINGER

PHILCO-FORD CORPORATION
WESTERN DEVELOPMENT LABORATORIES DIV.
3939 FABIAN WAY
PALO ALTO, CA 94303
  ATTN LIBRARY
  ATTN E. R. HAHN, MS-X22
  ATTN S. CRAWFORD, MS-31

PULSAR ASSOCIATES, INC.
7911 HERSCHEL AVENUE
LA JOLLA, CA 92037
  ATTN CARLTON JONES

RCA CORPORATION
GOVERNMENT AND COMMERCIAL SYSTEMS
MISSILE AND SURFACE RADAR DIVISION
MARNE HIGHWAY AND BORTON LANDING ROAD
MOORESTOWN, NJ 08057
  ATTN ELEANOR DALY

RCA CORPORATION
GOVERNMENT AND COMMERCIAL SYSTEMS
ASTRO ELECTRONICS DIVISION
P.O. BOX 800
PRINCETON, NJ 08540
  ATTN DR. GEORGE BRUCKER

RCA CORPORATION
DAVID SARNOFF RESEARCH CENTER
201 WASHINGTON ROAD
WEST WINDSOR TOWNSHIP
PRINCETON, NJ 08540
  ATTN WILLIAM J. DENNEHY

RCA CORPORATION
P.O. BOX 591
SOMERVILLE, NJ 08876
  ATTN DANIEL HAMPEL, ADV COMM LAB

RCA CORPORATION
CAMDEN COMPLEX
FRONT & COOPER STREETS
CAMDEN, NJ 08012
  ATTN E. VAN KEUREN, 13-5-2

R & D ASSOCIATES
P.O. BOX 3580
SANTA MONICA, CA 90403
  ATTN WILLIAM KARZAS
  ATTN R. R. SCHAEFER
  ATTN DR. WM R. GRAHAM
  ATTN ROBERT A. POLL
  ATTN S. CLAY ROGERS

RADIATION DIVISION OF HARRIS
  INTERTYPE
P.O. BOX 37
MELBOURNE, FL 32902
  ATTN JOHN H. TURNER
  ATTN E. V. ROOS, MS 16-156

RAND CORPORATION, THE
1700 MAIN STREET
SANTA MONICA, CA 90406
  ATTN C. M. CRAIN

RAYTHEON COMPANY
528 BOSTON POST ROAD
SUDBURY, MA 01776
  ATTN HAROLD L. FLESCHER
  ATTN D. R. JONES

RESEARCH TRIANGLE INSTITUTE
P.O. BOX 12194
RESEARCH TRIANGLE, NC 27709
  ATTN ENG & ENVIRON SCIENCES DIV,
    DR. MAYRANT SIMONS, JR.

SANDERS ASSOCIATES, INC.
95 CANAL STREET
NASHUA, NH 03060
  ATTN M. L. AITEL
  ATTN 1-6270, R. G. DESPATHY, SR PE

SANDIA LABORATORIES
  ATTN DOCUMENT CONTROL
P.O. BOX 5800
ALBUQUERQUE, NM 87115
  ATTN ORG 50, A. NARATH
  ATTN ORG 9353, R. L. PARKER
  ATTN ORG 5231, C. N. VITTITOE
  ATTN ORG 1426, J. A. COOPER
  ATTN TECHNICAL LIBRARY
  ATTN ORG 1935, J. E. GOVER
  ATTN J. D. APPEL
  ATTN J. W. KANE

SANDIA LABORATORIES
LIVERMORE LABORATORIES
  ATTN DOCUMENT CONTROL
P.O. BOX 969
LIVERMORE, CA 94550
  ATTN T. A. DELLIN
  ATTN K. A. MITCHELL, 8157
  ATTN G. OTEY, 8178
  ATTN J. L. WIRTH, 8340
  ATTN SUPERVISOR, LIBRARY DIV
  ATTN J. A. MOGFORD, DIV 8341

SCIENCE APPLICATION INC.
HUNTSVILLE DIVISION
2109 W CLINTON AVE
SUITE 700
HUNTSVILLE, AL 35805
  ATTN N. R. BYRN

SINGER-GENERAL PRECISION, INC.
1150 MCBRIDE AVENUE
LITTLE FALLS, NJ 07424
  ATTN ABRAHAM WITTELES, RADIATION
    EFFECTS SUPERVISOR, 3-5820

SPERRY RAND CORPORATION
SPERRY GYROSCOPE DIVISION
GREAT NECK, NY 11020
  ATTN PAUL MARRAFFINO, DEPT 4282

SPERRY RAND CORPORATION
UNIVAC DIVISION
DEFENSE SYSTEMS DIVISION
P.O. BOX 3525, MAIL STATION 1931
ST. PAUL, MN 55101
  ATTN DENNIS AMUNDSON, MS 5261
  ATTN J. A. INDA, MS 5451
  ATTN A. BROWN, MS 8931

SPERRY RAND CORPORATION
SPERRY FLIGHT SYSTEMS DIVISION
P.O. BOX 21111
PHOENIX, AZ 85002
  ATTN PAT DEVILLIER
  ATTN D. A. SCHOW, RM 140C

STANFORD RESEARCH INSTITUTE
333 RAVENSWOOD AVENUE
MENLO PARK, CA 94025
  ATTN MR. PHILLIP DOLAN
  ATTN MR. ARTHUR LEE WHITSON
  ATTN DR. ROBERT A. ARMISTEAD
  ATTN J. A. BAER, J1015
  ATTN J. CHOWN

STANFORD RESEARCH INSITUTE
306 WYNN DRIVE, NW
HUNTSVILLE, AL 35805
  ATTN SR RES ENG. M. MORGAN
  ATTN HAROLD CAREY
  ATTN WILLIAM DRUEN

SUNDSTRAND AVIATION
4751 HARRISON AVENUE
ROCKFORD, IL 61101
  ATTN DEPT 763SW, CURT WHITE

SYSTEMS, SCIENCE AND SOFTWARE, INC.
P.O. BOX 1620
LA JOLLA, CA 92037
  ATTN GLEN SEAY

SYSTRON-DONNER CORPORATION
200 MIGUEL ROAD
CONCORD, CA 94520
  ATTN HAROLD D. MORRIS

TEXAS INSTRUMENT, INC.
P.O. BOX 5474
DALLAS, TX 75222
  ATTN R & D PROJECT MANAGER,
    MR. DONALD J. MANUS, MS 72
  ATTN RADIATION EFFECTS PROGRAM MGR,
    MR. GARY F. HANSON

TRW SYSTEMS GROUP
ONE SPACE PARK
REDONDO BEACH, CA 90278
  ATTN MR. R. KINGSLAND
  ATTN MR. D. JORTNER
  ATTN MR. A. ANDERMAN
  ATTN B. SUSSHOLTZ
  ATTN TECH INFO CENTER/S-1930
  ATTN LILLIAN SINGLETARY, R1/2154
  ATTN DR. W. A. ROBINSON, R1/2028
  ATTN R. M. WHITMER
  ATTN R. MOLMUD
  ATTN JAMES GORDON

DISTRIBUTION (Cont'd)

TRW SYSTEMS GROUP
SAN BERNADINO OPERATIONS
P.O. BOX 1310
SAN BERNADINO, CA 92402
  ATTN J. M. GORMAN, MGR, WPN SYS ENG
  ATTN D. W. PUGSLEY
  ATTN E. W. ALLEN
  ATTN H. S. JENSEN
  ATTN J. E. DAHNKE
  ATTN R. H. KARCHER, MS 526/712

UNITED AIRCRAFT CORPORATION
HAMILTON STANDARD
BRADLEY INTERNATIONAL AIRPORT
WINDSOR LOCKS, CT 06069
 ATTN RAYMOND G. GIGUERE

UNITED AIRCRAFT CORPORATION
NORDEN DIVISION
HELEN STREET
NORWALK, CT 06851
  ATTN CONRAD CORDA

WESTINGHOUSE ELECTRIC CORPORATION
DEFENSE AND SPACE CENTER
DEFENSE AND ELECTRONICS SYSTEMS CENTER
P.O. BOX 1693
BALTIMORE, MD 21203
  ATTN HENRY P. KALAPACA, MS 3519

WESTINGHOUSE ELECTRIC CORPORATION
RESEARCH AND DEVELOPMENT CENTER
1310 BEULAH ROAD (CHURCHILL BOR)
PITTSBURGH, PA 15235
  ATTN WILLIAM E. NEWELL

WESTINGHOUSE ELECTRIC CORPORATION
ASTRONUCLEAR LABORATORY
P.O. BOX 1084
PITTSBURGH, PA 15236
  ATTN P. W. DICKSON

HARRY DIAMOND LABORATORIES
  ATTN McGREGOR, THOMAS, COL COMMANDING
       OFFICER/FLYER, I.N./LANDIS, P.E./
       SOMMER, N. /CONRAD E.E.
  ATTN CARTER, W.W.,DR., ACTING TECHNICAL
       DIRECTOR/MARCUS, S.M.
  ATTN KIMMEL, S., PIO
  ATTN CHIEF, 0021
  ATTN CHIEF, 0022
  ATTN CHIEF, LAB 100
  ATTN CHIEF, LAB 200
  ATTN CHIEF, LAB 300
  ATTN CHIEF, LAB 400
  ATTN CHIEF, LAB 500
  ATTN CHIEF, LAB 600
  ATTN CHIEF, DIV 700
  ATTN CHIEF, DIV 800
  ATTN CHIEF, LAB 900
  ATTN CHIEF, LAB 1000
  ATTN RECORD COPY, BR 041
  ATTN HDL LIBRARY 3 CY
  ATTN CHAIRMAN, EDITORIAL COMMITTEE 4 CY
  ATTN CHIEF, 047
  ATTN TECH REPORTS, 013
  ATTN PATENT LAW BRANCH, 071
  ATTN MCLAUGHLIN, P.W., 741
  ATTN CHIEF, 0024
  ATTN CHIEF, 280
  ATTN J. R. MILLETTA, 240 (25 CY)
  ATTN CHIEF 1020
  ATTN R. WONG, 1020
  ATTN CHIEF, 1030
  ATTN W. T. WYATT, JR., 1000
  ATTN J. BEILFUSS, 1000
  ATTN JOHN BOMBARDT, 1000
  ATTN ROBERT GRAY, 1000
  ATTN JANIS KLEBERS, 1000
  ATTN LEO LEVITT, 1000
  ATTN JOHN ROSADO, 240
  ATTN JOHN TOMPKINS, 330
  ATTN DAVID FINLEY, 1020
  ATTN ALFRED BRANDSTEIN, 1020
  ATTN EGON MARX, 1020
  ATTN JOHN INGRAM, 1020
  ATTN CHIEF, 240
  ATTN ART HAUSNER, 0025
  ATTN AMXDO-NP (4 CY)
  ATTN AMXDO-EM (4 CY)

163
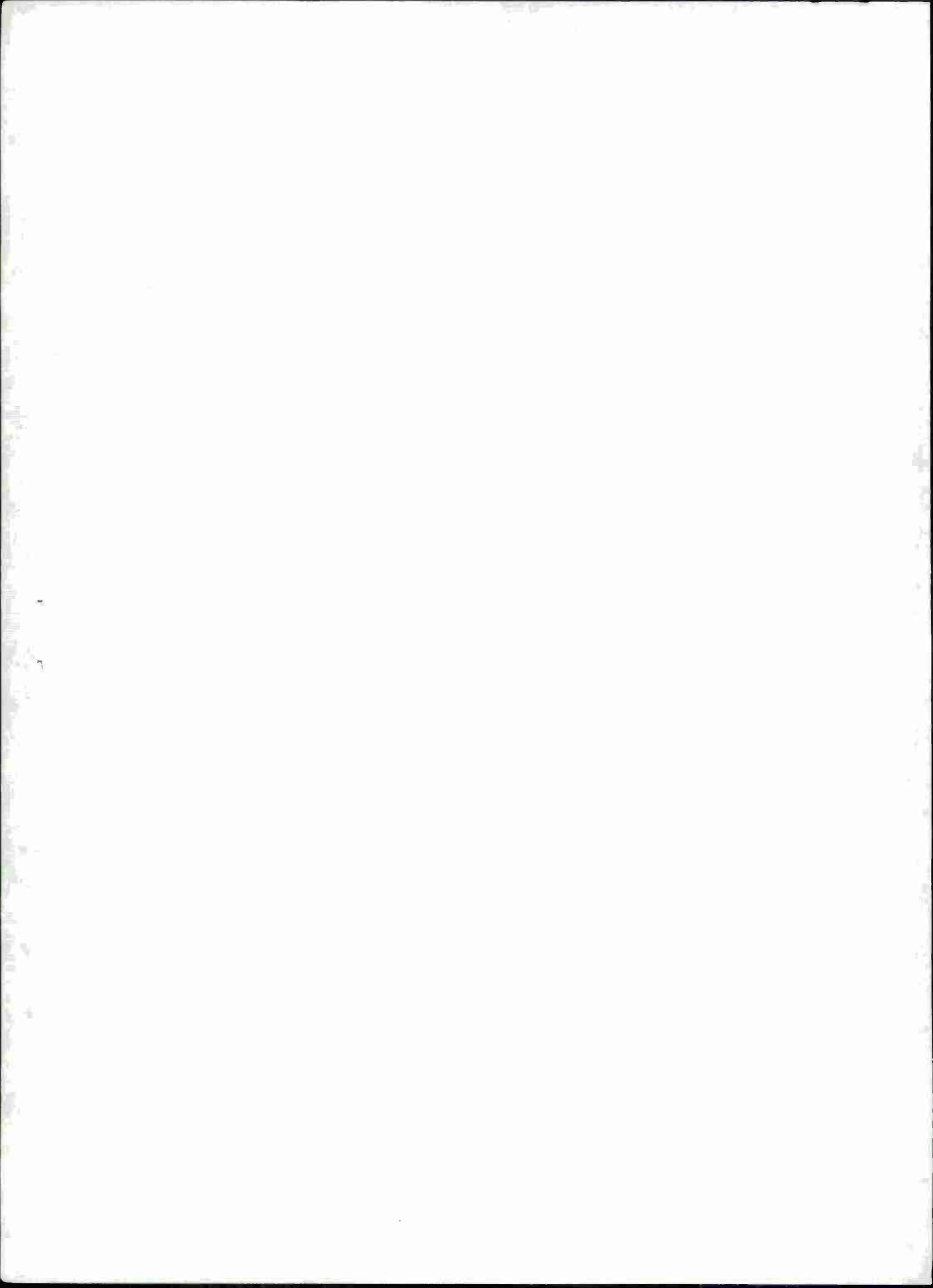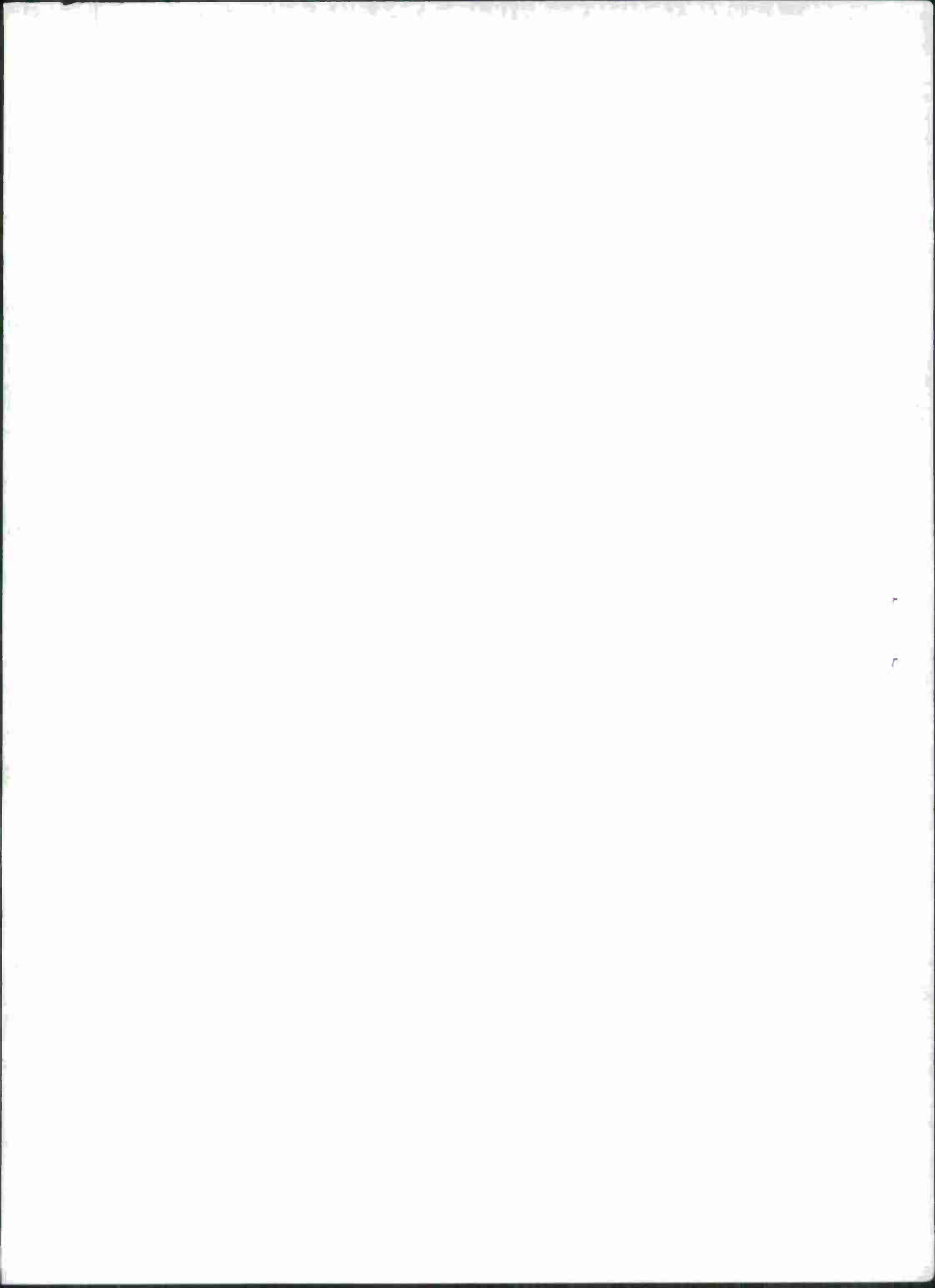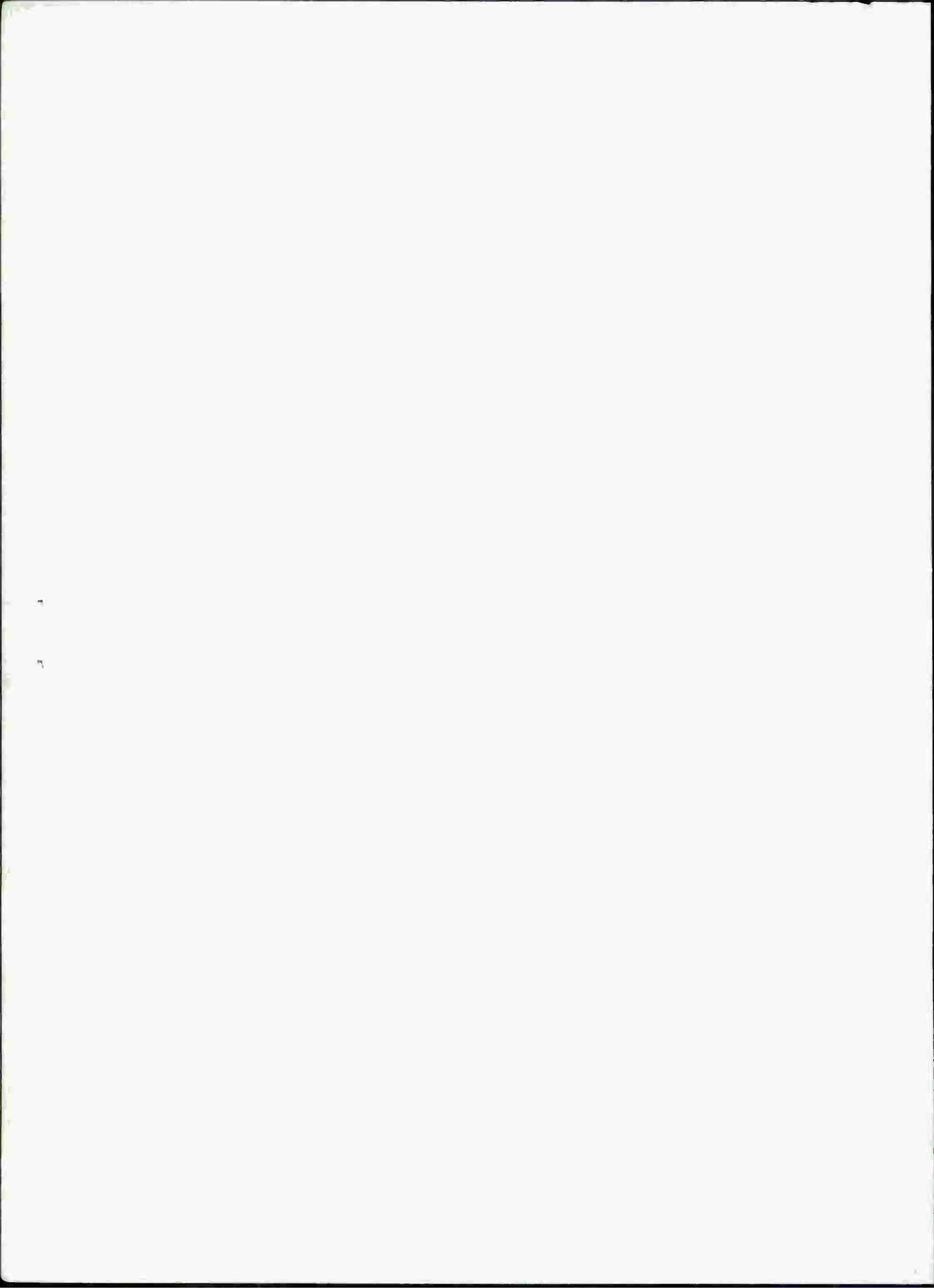
DEPARTMENT OF THE ARMY
HARRY DIAMOND LABORATORIES
2800 POWDER MILL RD
ADELPHI. MD 20783

AN EQUAL OPPORTUNITY EMPLOYER

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE $300

THIRD CLASS

SUPERINTENDENT
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA   93940
    ATTN CODE 2124, LIBRARY

2124

S

U1720